Fast Power Allocation Algorithms for Adaptive MIMO Systems

Jong-Sun Chung

A Thesis Submitted in Partial Fulfillment
of the Requirements for the Degree of

Master of Engineering
in
Electrical and Computer Engineering

University of Canterbury
Christchurch, New Zealand
September 14, 2009

Abstract

Recent research results have shown that the MIMO wireless communication architecture is a promising approach to achieve high bandwidth efficiencies. MIMO wireless channels can be simply defined as a link for which both the transmitting and receiving ends are equipped with multiple antenna elements. Adaptive modulation and power allocation could be used to further improve the performance of MIMO systems.

This thesis focuses on developing a fast and high performance power allocation algorithm. Three power allocation algorithms are proposed in this thesis and their performances are compared in various system sizes and transceiver architectures. Among the three algorithms proposed in this thesis, the fast algorithm may be considered as the best power allocation algorithm since the performance of the fast algorithm is almost as good as the fullsearch (optimal) algorithm and the mean processing time is considerably less than the fullsearch algorithm. The fast algorithm achieves about 97.6% agreement with the optimal throughput on average. In addition, the time taken to find the power scaling factors using the fullsearch algorithm is about 2300 times longer than the processing time of the fast algorithm in a 6×6 system when the SNR is 20dB.

As an extension to the power allocation process, excess power allocation methods are introduced. Excess power is the unused power during the power allocation process. The power allocation algorithm allocates power to each received SNR to maximize the throughput of the system whereas the excess

power allocation distributes the excess power to each SNR to improve both the instantaneous and temporal behavior of the system. Five different excess power allocation methods are proposed in this thesis. These methods were simulated in the Rayleigh fading channel with different Doppler frequencies, $f_D = 10$Hz, 50Hz and 100Hz, where the ACF of the channel coefficients are given by the Jakes' model. The equal BER improvement method showed a slightly better performance than the other methods. The equal BER improvement method enables the system to maintain the power scaling factors without sacrificing QoS for 19.6 ms on average when the maximum Doppler shift is 10Hz.

Acknowledgments

First and foremost, I am greatly indebted to my supervisor, Assoc. Prof. Peter J. Smith, for his continuous guidance, support, and encouragement throughout my master's research. He has continuously provided new ideas and patiently reviewed my thesis drafts. Without his endless support, I would never be able to complete this research.

Secondly, I would like to thank my family, my parents and my brother for their love, patience and continuous encouragement. I extend my gratitude to my friends and church members. Their encouragement and emotional support made my life full of warm and unforgettable moments.

Acronyms

ACF	autocorrelation function
AWGN	additive white Gaussian noise
BER	bit error rate
BPSK	binary phase shift keying
CSI	channel state information
i.i.d	independent and identically distributed
LOS	line-of-sight
MIMO	multiple-input multiple-output
MMSE	minimum mean square error
MPSK	M-ary PSK
MQAM	M-ary QAM
NLOS	non line-of-sight
QAM	quadrature amplitude modulation
QoS	quality of service
RX	receiver
SINR	signal-to-interference-plus-noise ratio
SISO	single-input single-output
SNR	signal-to-noise ratio
SVD	singular value decomposition
TX	transmitter
WF	water-filling

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Background

Over the last century, communication techniques have developed rapidly, especially in the wireless communication area. Wireless communication has grown from a relatively obscure service to a ubiquitous technology. According to the International Telecommunications Union, a UN agency, in 2008, around 4.1 billion people use a mobile phone [2]. That is approximately half of the world's population. Hence, the interest in wireless communication is rapidly growing.

In a communication system, two primary resources are employed: transmit power and channel bandwidth. The transmitted power is the average power of the transmit signal. The channel bandwidth is defined as the band of frequencies allocated for transmission of the message signal [3]. Thus, the general objective of system design is to maximize the efficiency with which the two resources are used.

Recent research has shown that the multiple-input multiple-output (MIMO) architecture is a promising approach to achieve highly efficient wireless communication systems [4], [5]. MIMO is a wireless communication technique that uses N_t transmit antennas to transmit messages and the receiver receives the message with N_r receive antennas to improve the link quality and the capacity

of the system. As shown in [4] and [5], the capacity of the system increases linearly with the number of antenna elements.

However, the performance of the system does not depend only on the system size but also on the channel environment, transceiver architecture, target bit error rate (BER), etc. This thesis considers three commonly used wireless channel models: Rayleigh, Rician, and spatially correlated Rayleigh fading channels. Rayleigh fading is commonly used to model scattering or multipath fading when there is no dominant propagation along a direct line-of-sight (LOS) between the transmitter (TX) and the receiver (RX). When there exists a strong LOS component, then Rician fading is used to model the channel. The Rician fading channel consists of both a LOS component and a non-LOS (NLOS) component. Spatially correlated Rayleigh fading is used to model the scenario when there exists correlation between the received signals.

As the demand for MIMO increases, intensive research has been carried out to develop reliable and efficient transceivers. It is well known that the singular value decomposition (SVD) transceiver architecture provides relatively high throughput compared to the minimum mean square error (MMSE) system. SVD systems require a feedback link since channel state information (CSI) must be known at both TX and RX in order to perform the SVD processing. In contrast, MMSE systems do not require CSI knowledge at the TX. Thus, SVD systems have higher complexity relative to MMSE systems.

Moreover, many researchers have attempted to find techniques to combine with MIMO to maximize the potential benefits of MIMO [6], [7], [8]. One of these techniques is adaptive modulation. Adaptive modulation is a technique wherein modulation constellations are dynamically adapted in accordance with channel conditions while providing the required quality of service (QoS) [9]. Power allocation is another technique that is used in wireless communication systems to maximize the throughput of the system. The power allocation process allocates different amounts of transmit power to each transmit antenna.

Hence, power can be reduced where it is not efficiently used and allocated to other antennas in order to improve the performance of the system. Thus, power allocation together with adaptive modulation is a powerful approach to increasing throughput under power and QoS constraints.

1.2 Motivation

Commonly used wireless communication techniques used in conjunction with the MIMO approach are adaptive modulation and power allocation. The standard bit error rate (BER) formulae for a given signal-to-noise ratio (SNR) for M-ary phase shift keying (MPSK) and M-ary quadratue amplitude modulation (MQAM) are given in [10]. Thus, if the system uses no transmission, binary PSK, 4QAM, 16QAM, and 64QAM modulation modes to transmit data, as assumed throughout the thesis, we could compute the required SNR or the threshold to satisfy the target BER for each modulation scheme. In other words, the exact moment to switch from one modulation scheme to another is known. However, there is no simple rule for the power allocation. Thus, allocating power to a channel in a computationally efficient way is a very important technique. Note that no transmission implies that a particular eigen-channel is not used. However, all antennas are in operation and the remaining eigen-channels are in operation. In this thesis, we concentrate on developing fast and efficient power allocation algorithms. By giving a certain amount of extra power to a channel, the modulation scheme of the channel could be upgraded to a higher modulation mode. However, in order to provide the extra power, we must reduce some power from another channel since the power is limited. Hence, the aim of the power allocation scheme is to find the optimal combination of the power scaling factors to achieve the maximum throughput. Three new power allocation algorithms are proposed in this thesis. The power allocation algorithms assume the average transmit power is one. By varying the

noise power, σ^2, this gives us a general SNR value of $1/\sigma^2$. Hence, the units are unimportant.

The optimal power allocation algorithm introduced in this thesis is the fullsearch algorithm, which guarantees to find the ideal power scaling factors. The fullsearch algorithm considers all possible combinations of power scaling factors, which consequently increases the processing time. The fullsearch algorithm computes scaling factors to scale to each modulation mode for each received SNR. Thus, it picks the optimal power scaling factors among the N^m combinations of the power scaling factor where N is the number of modulation modes and m is the number of channels (SNRs). The computational workload increases by a huge amount, as the system size gets larger. The high computational workload makes this algorithm almost inapplicable for systems larger than 2×2 system. Note that 2×2 denotes a $N_r \times N_t$ system. Therefore, approximate algorithms have been developed instead: the fast algorithm and the modified fast algorithm.

The fast and the modified fast algorithm reduce the power allocation processing time significantly while achieving, on average, around 97.6% and 99.3% agreement with the fullsearch (optimal) results respectively. The advantage of the fast processing time of the fast and the modified fast algorithm becomes more valuable as the system size gets larger or as the SNR increases. The fast algorithm is faster relative to the modified fast algorithm but the performance is slightly inferior.

The power allocation process does not consume all of the available power since the power allocation algorithms provide just enough power to each SNR to scale the SNR to the threshold of a modulation mode. Hence, an efficient algorithm to use the spare power is also required. The unused power, called the excess power, can be used to improve both the system's temporal and instantaneous behavior. We propose five different methods to allocate the excess power in the thesis.

As power allocation is known to be an important component of high throughput the MIMO systems, many power allocation algorithms are explored in the literature. The most well-known power allocation method is the water-filling (WF) method. The water-filling method given in [11] is known to be the optimal power distribution method which utilizes the SVD technique. However, the WF solution requires an infinite-length codebook, a continuous modulation order and a continuous power level. Thus, it is not possible to use the result directly in practice.

The modified power allocation method, called QoS based WF, is given in [12]. It assumes that perfect CSI is known at the TX as is also assumed in this thesis. The QoS based WF algorithm scales down each SNR to the threshold level of the corresponding modulation mode and then distributes the power in the order from the strongest to the weakest. However, giving the power in this order may not be the optimal solution, especially when two active channels are using the same modulation mode. An improved power scaling solution in this situation is given by the modified fast algorithm in Chapter 3. Moreover, the modified fast algorithm is relatively simple compared to the QoS based WF since QoS based WF uses a complicated function which can only be solved numerically.

In [6], [8] and [13], power allocation methods in imperfect CSI are proposed. In [13], a greedy power allocation algorithm is proposed and is proven to be better than traditional algorithms such as water-filling and equal power allocation algorithms. The greedy power allocation can mitigate the effects of imperfect CSI and strike a good balance between the system throughput and the target BER. This algorithm scales each SNR to no transmission mode and computes the required power to scale each SNR to the next modulation mode. One SNR, which requires the minimum power to scale up to the next modulation mode, is scaled up at each iteration. This process is repeated scaling up one SNR at a time until all the power has been allocated. Consequently, the

number of iterations increases as the number of modulation modes increases. This iterative method is relatively slow and has a fairly high complexity. In contrast, the fast algorithm proposed in this thesis is simple and relatively fast, only requiring the same number of iterations as the number of channels.

Most published papers regarding power allocation concentrate on maximizing the instantaneous system throughput under QoS and power constraints. It is interesting that the use of the excess power to improve the instantaneous and temporal behavior is barely mentioned in the literature.

1.3 Thesis Outline

This thesis presents new power allocation algorithms that could be used in MIMO wireless systems and five new excess power allocation methods. We consider various system models. Throughout this thesis, we assume the average transmit power is unity and the modulation modes used in adaptive modulation are no transmission, BPSK, 4QAM, 16QAM, and 64QAM.

Chapter 2 introduces the theoretical background of the mobile propagation channel and MIMO systems. Some common statistical channel models such as the Rayleigh channel, the Rician channel, and spatially correlated Rayleigh channels are elaborated. Then, a detailed analysis of the SVD and the MMSE systems are given. Chapter 2 also includes the basics of adaptive modulation and a brief introduction to power allocation at the end of the chapter.

Chapter 3 focuses on developing the power allocation algorithms. This chapter begins with a thorough explanation of power allocation. After the explanation, three power allocation algorithms are presented with a numerical example for each algorithm. At the end of Chapter 3, the performance comparisons are provided. In particular, the mean processing time taken by each algorithm to compute the power scaling factors are given for 2×2, 4×4, and 6×6 system sizes. In addition, mean throughput comparisons between

the SVD system and the MMSE system are given. Furthermore, the mean throughput of the Rayleigh, the Rician, and the spatially correlated Rayleigh channels are given for various system parameters.

Chapter 4 illustrates five different methods to allocate the excess power. The excess power is added to the power scaling factors found by the power allocation algorithm to improve both the temporal and instantaneous behavior of the system. The equal increment method scales the received SNRs by an equal amount above the threshold. The equal BER improvement method uses the excess power to equally improve the BER performance of the received SNRs. By considering the fact that the more variable SNRs may change more quickly [1], the eigenvalue variance method distributes the excess power proportional to the eigenvalue variances. Another method was developed to allocate the excess power proportional to the received SNR. The final method proposed is to allocate the excess power proportional to the difference between the current SNR and the SNR that was received at τ seconds before. Comparisons between the methods are provided at the end of the chapter.

Chapter 5 gives some conclusions, and some future research directions are pointed out.

Chapter 2

System Model

2.1 Multiple-input Multiple-output Channel

MIMO is a wireless communication technique that uses multiple antenna elements at both link ends as shown in Fig. 2.1. MIMO improves the link quality, and the capacity of the system increases linearly with the number of antenna elements [4], [5], [14].

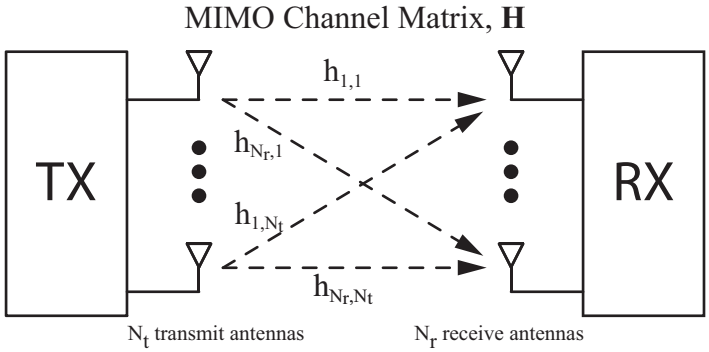


Figure 2.1: Block diagram of a MIMO system.

In MIMO systems, a transmitter with N_t transmit antennas transmits multiple streams of data. The streams propagate through the matrix channel, **H**, which consists of multiple paths between the transmitter (TX) and the receiver (RX). Then, the receiver receives a signal vector with N_r receive antennas.

Hence, we can model the MIMO system mathematically as:

$$\mathbf{r} = \mathbf{H}\mathbf{s} + \mathbf{n} \tag{2.1}$$

where $\mathbf{r}$, $\mathbf{s}$ and $\mathbf{n}$ are the receive, transmit and channel noise vectors, respectively. $\mathbf{H}$ is the $N_r \times N_t$ channel matrix and is denoted as:

$$\mathbf{H} = \begin{pmatrix} h_{1,1} & h_{1,2} & \cdots & h_{1,N_t} \\ h_{2,1} & h_{2,2} & \cdots & h_{2,N_t} \\ \vdots & \vdots & \ddots & \vdots \\ h_{N_r,1} & h_{N_r,2} & \cdots & h_{N_r,N_t} \end{pmatrix} \tag{2.2}$$

The entries, $h_{i,j}$, are the transfer functions from the jth transmit antenna to the ith receive antenna as depicted in Fig. 2.1.

MIMO systems are commonly used in high-rate wireless communications because of their high spectral efficiency and robustness to multipath fading. This thesis considers MIMO systems used in Rayleigh, Rician and spatially correlated channels.

2.2 Channel Models

2.2.1 Rayleigh Channel Model

The Rayleigh distribution is commonly used to model scattering or multipath fading when no direct line-of-sight (LOS) path is available. By the central limit theorem, the channel can be modeled as a Gaussian process. Hence, the channel coefficients in (2.2) can be modeled as

$$h_{r,s} = X + jY \tag{2.3}$$

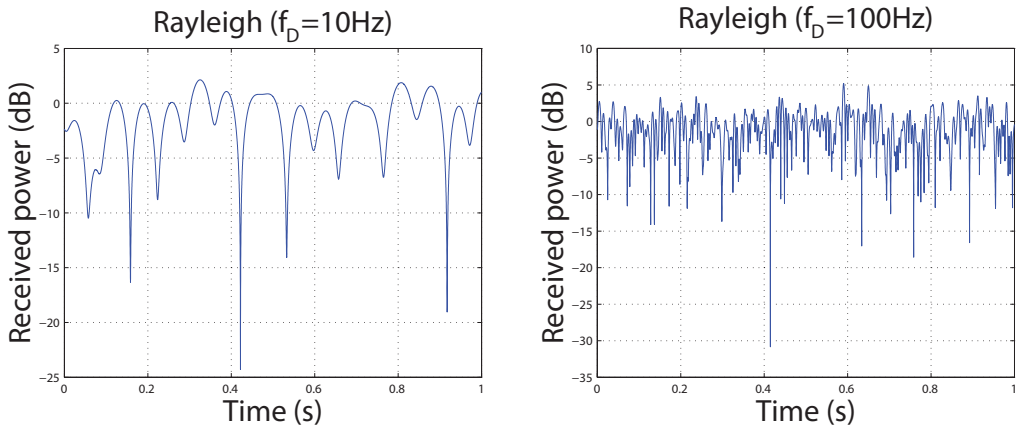
where X and Y are independent and identically distributed (i.i.d) $\mathcal{N}(0, \frac{1}{2})$. The notation $\mathcal{N}(\mu, \sigma^2)$ is used to represent a real Gaussian random variable with mean μ and variance σ^2.

In mobile communication systems, the velocity of the receiver and/or transmitter affects how rapidly the channel fades. The receiver receives a Doppler shifted signal caused by the motion. The Doppler frequency, f_D (Hz), is given by

$$f_D = \frac{v f_c}{c}, \tag{2.4}$$

where v, f_c and c are the velocity of the mobile station, carrier frequency and the speed of light respectively.

The power variations of a constant signal passing through a Rayleigh fading channel over one second is shown in Fig. 2.2. As the relative velocity between the transmitter and the receiver increases, the channel varies more rapidly and it is worth noting that the strength of the signal can drop by a factor of several hundreds within half a second.



(a) Maximum Doppler shift of 10Hz. (b) Maximum Doppler shift of 100Hz.

Figure 2.2: One second of Rayleigh fading generated using the Jakes' model.

To model the temporal behavior of the Rayleigh fading channel, the Jakes' model is considered. This model assumes the incoming signals at the moving receiver are isotropic [1]. In other words, the scatterers are uniformly distributed around a circle. In this case, the autocorrelation function (ACF) of

the channel coefficient is given by [15]

$$R(\tau) = J_0(2\pi f_D \tau) \tag{2.5}$$

where $J_x(\cdot)$ represents the xth order Bessel function of the first kind, f_D is the Doppler frequency (maximum Doppler shift) and τ is the time displacement.

The Rayleigh fading channel generated using Jakes' model is shown in Fig. 2.2. Throughout this research, this model is used to simulate the temporal behavior of the Rayleigh fading channel.

2.2.2 Rician Channel Model

The Rician distribution is used to model the channel when a LOS propagation path between transmitter and receiver is available unlike the Rayleigh distribution. The channel coefficients, $h_{i,j}$, of the Rician channel matrix, $\mathbf{H}_{\text{rice}}$, can also be modeled as in (2.3) where X and Y are i.i.d $\mathcal{N}(\mu, \sigma^2)$. Note that the mean is no longer zero due to the LOS component. The Rician fading channel consists of two distinct components:

i. a specular component that illuminates the entire array and is thus spatially deterministic from antenna to antenna (LOS component);
ii. a scattered Rayleigh distributed component which varies randomly from antenna to antenna (non-LOS (NLOS) component).

The Rician distribution is characterized by a Rician factor, K, which is defined as the ratio of deterministic-to-scattered power. The standard MIMO Rician fading channel model is given by [16]

$$\mathbf{H}_{\text{rice}} = \sqrt{\frac{K}{K+1}}\mathbf{H}_{LOS} + \sqrt{\frac{1}{K+1}}\mathbf{H}_{NLOS} \tag{2.6}$$

where K is the Rician factor, $\mathbf{H}_{LOS}$ and $\mathbf{H}_{NLOS}$ are the unit power specular and the scattered components respectively.

Note that the scattered component, $\mathbf{H}_{NLOS}$, is equivalent to the Rayleigh i.i.d channel matrix. As the Rician factor, K, approaches zero, in other words, as the LOS signal power decreases towards zero, the channel matrix, $\mathbf{H}_{\text{rice}}$, eventually approaches the Rayleigh fading case.

Due to the reduced rank behavior of the Rician channel, which degrades the quality of the links, the achievable capacity for spatial multiplexing systems in the Rician channel is relatively low compared to systems in i.i.d Rayleigh environments under the same SNR, especially when the transmit power is uniformly distributed among the TX antennas [17]. Analytical results on system capacity for a MIMO Rician channel can be found in [18].

For the simulation of the Rician channel throughout this research, the channel matrix, $\mathbf{H}_{\text{rice}}$, is generated according to the model given in (2.6). The specular component, $\mathbf{H}_{LOS}$, is kept constant and the scattered component, $\mathbf{H}_{NLOS}$, is generated in the same way as the i.i.d Rayleigh channel. The entries of $\mathbf{H}_{NLOS}$ are independent and each forms a stationary Gaussian process over time with zero mean and ACF defined by (2.5).

2.2.3 Spatially Correlated Rayleigh Fading Channel Model

In many practical situations sparse scattering and insufficient spacing between adjacent antennas can cause correlation between the received signals. The spatial correlations degrade both capacity and bit error rate (BER) performance [19], [20]. The correlations between antennas are mainly governed by three parameters [21]: the distance between antennas, the angular spread of the arrival incident waves, and the mean angle of arrival of the incident waves.

With the assumption that the correlation among the receive antennas is independent of the correlations between the transmit antennas, the standard model of a Rayleigh fading channel with spatial correlation at both TX and

RX is given by [22]

$$\mathbf{H}_{\text{corr}} = \mathbf{R}_{\text{rx}}^{1/2}\ \mathbf{H}_{\text{i.i.d}}\ \mathbf{R}_{\text{tx}}^{1/2} \tag{2.7}$$

where $\mathbf{H}_{\text{i.i.d}}$ is a $N_r \times N_t$ matrix whose entries are i.i.d $\mathcal{CN}(0,1)$ as in (2.3). The notation $\mathcal{CN}(\cdot)$ is used to represents a complex Gaussian random variable. The matrices $\mathbf{R}_{\text{tx}}$ and $\mathbf{R}_{\text{rx}}$ are the transmit and receive correlation matrices with dimensions $N_t \times N_t$ and $N_r \times N_r$, respectively. When there are no correlations at either TX or RX, i.e. $\mathbf{R}_{\text{tx}}$ and $\mathbf{R}_{\text{rx}}$ are identity matrices, then the correlated channel matrix, $\mathbf{H}_{\text{corr}}$, reduces to $\mathbf{H}_{\text{i.i.d}}$ which is the same as the i.i.d Rayleigh channel.

A simple and convenient model for the correlation matrices, $\mathbf{R}_{\text{tx}}$ and $\mathbf{R}_{\text{rx}}$, is given by [22] as

$$\mathbf{R}_{\text{tx}} = \begin{pmatrix} 1 & \alpha & \cdots & \alpha^{(N_t-1)} \\ \alpha & 1 & \cdots & \alpha^{(N_t-2)} \\ \vdots & \vdots & \ddots & \vdots \\ \alpha^{(N_t-1)} & \alpha^{(N_t-2)} & \cdots & 1 \end{pmatrix} \tag{2.8}$$

$$\mathbf{R}_{\text{rx}} = \begin{pmatrix} 1 & \beta & \cdots & \beta^{(N_r-1)} \\ \beta & 1 & \cdots & \beta^{(N_r-2)} \\ \vdots & \vdots & \ddots & \vdots \\ \beta^{(N_r-1)} & \beta^{(N_r-2)} & \cdots & 1 \end{pmatrix} \tag{2.9}$$

where α and β are the spatial correlation coefficients at the transmitter and receiver respectively. These correlation coefficients vary between zero and 1 where $\alpha, \beta = 0$ indicates that the channels are uncorrelated and $\alpha, \beta = 1$ means that the channels are completely correlated.

As for the Rician channel, MIMO systems with spatial correlations are said to be "rank-deficient", as there may be only a few dominant eigenmodes while the other eigenvalues are relatively weak. In spite of the degradation of system throughput due to spatial correlation, the achievable performance is still relatively high in comparison with systems with a single antenna [1], [23].

For the simulation of the spatially correlated Rayleigh fading channel throughout this research, the channel matrix, $\mathbf{H}_{\text{corr}}$, is generated according to the channel model given in (2.7) with the correlation matrices, $\mathbf{R}_{\text{tx}}$ and $\mathbf{R}_{\text{rx}}$, fixed. The i.i.d channel matrix, $\mathbf{H}_{\text{i.i.d}}$, is the only time varying component in this model. The entries of $\mathbf{H}_{\text{i.i.d}}$ are independent and each forms a stationary Gaussian process over time with zero mean and ACF defined by (2.5).

2.3 Singular Value Decomposition

The singular value decomposition (SVD) is a very powerful tool to analyze MIMO systems. The SVD approach leads to a straightforward transmission architecture where the MIMO channel matrix is decomposed into parallel single-input single-output (SISO) sub-channels with non-equal gains as shown in Fig. 2.3 [5], [24]. In [25], the architecture combining power allocation and the SVD approach is considered as a special case of optimal precoder and decoder design for a MIMO channel. However, to perform SVD, perfect channel state information (CSI) is required. Throughout this research, we are going to assume that the perfect CSI is known.

The channel matrix, $\mathbf{H}$, can be expressed as the following by the SVD.

$$\mathbf{H} = \mathbf{U}\mathbf{D}\mathbf{V}^{\dagger}, \tag{2.10}$$

where $\mathbf{U} \in \mathbb{C}^{N_r \times N_r}$ and $\mathbf{V} \in \mathbb{C}^{N_t \times N_t}$ are unitary, and $(\cdot)^{\dagger}$ represents the complex conjugate transpose. $\mathbf{D} \in \mathbb{R}^{N_r \times N_t}$ is a non-negative diagonal matrix. The diagonal elements $\sqrt{\lambda_1} \geq \sqrt{\lambda_2} \geq \cdots \geq \sqrt{\lambda_m}$ of $\mathbf{D}$ are the square roots of the eigenvalues of $\mathbf{H}\mathbf{H}^{\dagger}$ where $m = \min(N_t, N_r)$ which is the rank of $\mathbf{H}$. The columns of $\mathbf{U}$ are the eigen-vectors of $\mathbf{H}\mathbf{H}^{\dagger}$ and the columns of $\mathbf{V}$ are the eigen-vectors of $\mathbf{H}^{\dagger}\mathbf{H}$. Thus, the MIMO system model in (2.1) can be written as

$$\mathbf{r} = \mathbf{U}\mathbf{D}\mathbf{V}^{\dagger}\mathbf{s} + \mathbf{n} \tag{2.11}$$

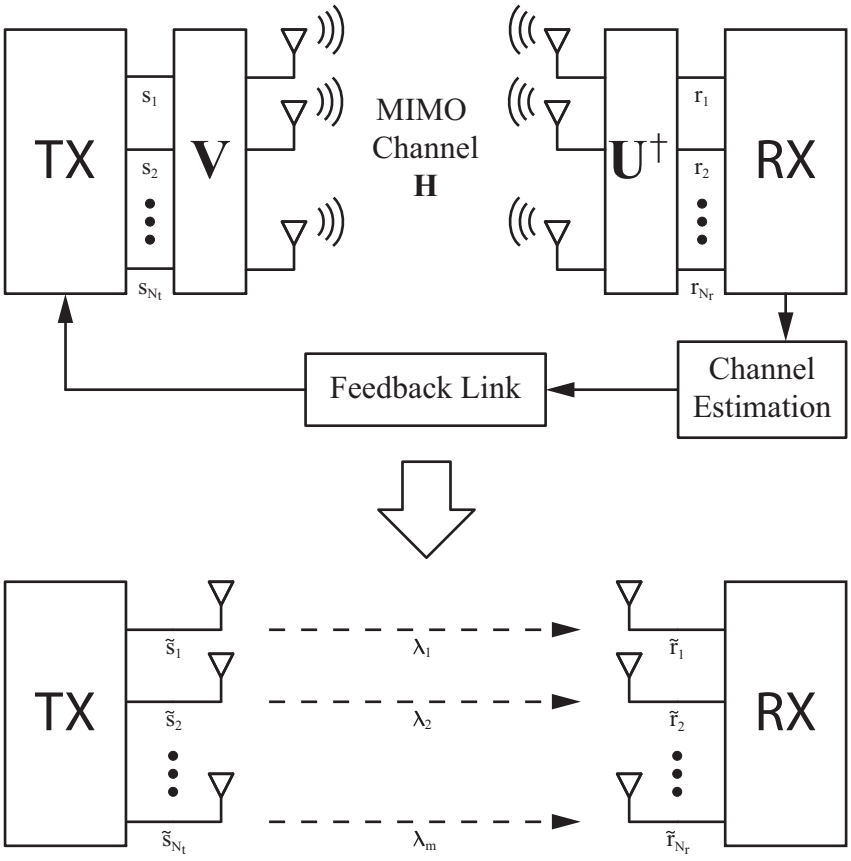


Figure 2.3: The MIMO transceiver architecture based on the SVD, which transforms the channel matrix into a bank of scalar links ($m = \min(N_t, N_r)$).

Let $\widetilde{\mathbf{r}} = \mathbf{U}^{\dagger}\mathbf{r}$, $\widetilde{\mathbf{s}} = \mathbf{V}\mathbf{s}$ and $\widetilde{\mathbf{n}} = \mathbf{U}^{\dagger}\mathbf{n}$. The transformation of the noise components does not change its statistical properties, so $\widetilde{\mathbf{n}}$ has the same distribution as $\mathbf{n}$, i.e. i.i.d complex Gaussian random variables with zero mean. Hence, the system model in (2.11) can be written as

$$\widetilde{\mathbf{r}} = \mathbf{D}\widetilde{\mathbf{s}} + \widetilde{\mathbf{n}} \tag{2.12}$$

Furthermore, (2.12) can be written component-wise as

$$\widetilde{r}_i = \sqrt{\lambda_i}\widetilde{s}_i + \widetilde{n}_i, \qquad 1 \leq i \leq \min(N_r, N_t) \tag{2.13}$$

Thus, if we pre-filter the input message symbols, $\mathbf{s}$, by the matrix $\mathbf{V}$ and post-filter the received symbols, $\mathbf{r}$, by the matrix $\mathbf{U}^{\dagger}$ as depicted in Fig. 2.3, the ordered eigenvalues of the channel can be realized as the power gains of m independent spatial channels, also known as the eigenmodes [1].

By assuming the average transmit power is 1, $\mathrm{E}|s_i|^2 = 1$, the signal-to-noise ratio (SNR) of the ith channel is defined as

$$\mathrm{SNR}_{(\mathrm{svd})i} = \frac{\lambda_i}{\sigma_n^2} \tag{2.14}$$

where σ_n^2 is the noise power, $\mathrm{E}|n_i|^2 = \sigma_n^2$.

Note that the received SNRs in a MIMO system with SVD transceiver architecture are relatively high compared to a MIMO minimum mean square error (MMSE) receiver system. However, since the SVD systems require a feedback link, the work load is higher than the MMSE systems.

2.4 Minimum Mean Square Error Receiver

A minimum mean square error (MMSE) receiver is a very well-known linear receiver with low complexity. A MMSE receiver minimizes the expected mean square error between the transmit symbol, $\mathbf{s}$, and its estimate, $\widetilde{\mathbf{s}}$, detected at

the receiver, thereby providing a better balance between interference mitigation and noise enhancement [26]. In MMSE, the optimization problem can be stated mathematically as

$$\begin{aligned} &\text{minimize} \quad \mathrm{E}|\mathbf{s}-\widetilde{\mathbf{s}}|^2 \\ &\text{subject to} \quad \widetilde{\mathbf{s}} = \mathbf{W}^{\dagger}\mathbf{r} \end{aligned} \tag{2.15}$$

where $\mathbf{W}$ is the MMSE filter coefficient matrix. The filter coefficient matrix, $\mathbf{W}$, is obtained by solving (2.15). As shown in [27], the optimal solution, assuming $\mathrm{E}|s_i|^2 = 1$, is given by

$$\mathbf{W} = \left(\sigma_n^2 \mathbf{I}_{N_t} + \mathbf{H}^{\dagger}\mathbf{H}\right)^{-1} \mathbf{H}^{\dagger}, \tag{2.16}$$

where σ_n^2 is the noise power and $\mathbf{I}_{N_t}$ is the $N_t \times N_t$ identity matrix.

In MMSE, the received signal is corrupted due to the multipath propagation and the interference between the antennas. Thus, in general, the signal-to-interference-plus-noise ratio (SINR) is used to measure the received signal strength [28]. In this thesis, $\mathrm{SINR}_{(\mathrm{mmse})}$ is used to represent the SINR in a MMSE system. The SINR on the ith spatial stream can be expressed as [29]

$$\mathrm{SINR}_{(\mathrm{mmse})i} = \mathbf{h}_i^{\dagger} \left(\sigma_n^2 \mathbf{I} + \mathbf{H}_{(-i)}\mathbf{H}_{(-i)}^{\dagger}\right)^{-1} \mathbf{h}_i \tag{2.17}$$

where $\mathbf{h}_i \in \mathbb{C}^{\,N_r \times 1}$ is the ith column of the channel matrix, $\mathbf{H}$, and $\mathbf{H}_{(-i)} \in \mathbb{C}^{\,N_r \times N_t - 1}$ is the $\mathbf{H}$ matrix with the ith column removed.

The received SNRs in a MIMO system with a MMSE receiver are relatively low compared to the SVD approach. However, the advantage of a MMSE receiver is that it is simple to implement and the work load is relatively low compared to the SVD system since the CSI is not required at TX side.

2.5 Adaptive Modulation

Adaptive modulation, also called bit loading [9], is a technique that is used in wireless communication systems to increase the system throughput. The

idea is to switch the modulation mode of the transmit data according to the SNR of the channel while maintaining the BER above the specified BER of the system. Adaptive modulation together with power allocation can increase the system throughput considerably [30].

In adaptive modulation, we assume that the transmit power is uniformly distributed over the transmit antennas. The modulation modes used are: no transmission (off), binary phase shift keying (BPSK), and higher order square quadrature amplitude modulations (QAM) up to 64QAM as in the general adaptive modulation approach [9].

As the modulation mode varies according to the SNR of the channel, the threshold SNR at which the system changes its modulation mode must be defined in terms of the target BER, the maximum BER that the system can tolerate. In other words, the minimum SNR that can guarantee the quality of service (QoS) of the link with the chosen modulation modes must be known.

The BER expression for BPSK in additive white Gaussian noise (AWGN) channel is commonly expressed as [10]

$$\mathrm{BER}_{\mathrm{BPSK}} \approx \frac{1}{2}\mathrm{erfc}\left(\sqrt{\mathrm{SNR}}\right). \tag{2.18}$$

Hence, by making the SNR the subject, the threshold SNR expression is obtained in terms of the target BER as the following.

$$\mathrm{SNR}_{\mathrm{BPSK}} \approx \left(\mathrm{erfc}^{-1}\left(2\mathrm{BER}_{\mathrm{target}}\right)\right)^2. \tag{2.19}$$

As for the square M-ary QAM (MQAM) with Gray mapping, the commonly used BER expression is [10]

$$\mathrm{BER}_{\mathrm{MQAM}} \approx \frac{2}{k}\left(1-\frac{1}{\sqrt{M}}\right)\mathrm{erfc}\left(\sqrt{\frac{3}{2}\frac{\mathrm{SNR}}{M-1}}\right), \tag{2.20}$$

where k is the number of bits per symbol, $M = 2^k$ is the constellation size and $\mathrm{erfc}(\cdot)$ is the complementary error function. Inverting (2.20) gives

$$\mathrm{SNR}_{\mathrm{thresh}} \approx \frac{1}{\beta}\left(\mathrm{erfc}^{-1}\left(\frac{\mathrm{BER}_{\mathrm{target}}}{\alpha}\right)\right)^2. \tag{2.21}$$

Modulation Modes	Threshold SNR	Threshold SNR (dB)
BPSK	4.7748	6.7895
4 QAM	9.5495	9.7998
16 QAM	45.1128	16.5430
64 QAM	179.8460	22.5490

Table 2.1: Threshold SNR for each modulation mode at $\mathrm{BER}_{\mathrm{target}} = 1 \times 10^{-3}$.

where

$$\alpha = \frac{2}{k}\left(1 - \frac{1}{\sqrt{M}}\right), \tag{2.22}$$

and

$$\beta = \frac{3}{2\left(M-1\right)}. \tag{2.23}$$

Hence, the threshold SNRs for each modulation mode are defined as

$$\mathrm{SNR}_{\mathrm{thresh}} = \begin{cases} \left(\mathrm{erfc}^{-1}\left(2\mathrm{BER}_{\mathrm{target}}\right)\right)^2, & \text{for } \; k = 1 \\ \frac{1}{\beta}\left(\mathrm{erfc}^{-1}\left(\frac{\mathrm{BER}_{\mathrm{target}}}{\alpha}\right)\right)^2, & \text{for } \; k = 2, 4, 6 \end{cases} \tag{2.24}$$

For example, the threshold SNR values in dB scale are given in Table 2.1 in the case of $\mathrm{BER}_{\mathrm{target}} = 1 \times 10^{-3}$. If the received SNR of a channel is 10dB, then the system uses 4QAM to transmit data and if, in the next instance, the SNR drops to 9dB, then the system switches its modulation mode to BPSK. Thus, without wasting power or sacrificing BER, these schemes provide a high average spectral efficiency [10].

2.6 Power Allocation

Power allocation is an another technique that is used in wireless MIMO systems to maximize the number of bits transmitted. In other words, it maximizes the throughput under power and QoS constraints. As briefly metioned in section 2.5, adaptive modulation together with power allocation can increase the system capacity considerably. Results from [30] indicates that the Shannon capacity of a flat fading channel is achieved by varying both transmission

rate and power. If adaptive modulation deals with transmission rate, power allocation considers the transmission power across the transmit antennas.

As in the example at the end of section 2.5, the system uses BPSK to transmit data when the SNR of the channel drops to 9dB. However, from Table 2.1, the required SNR for BPSK transmission mode is 6.7985dB to provide BER $\leq 1 \times 10^{-3}$. Thus, 9dB is extravagant for BPSK transmission.

The basic idea of power allocation is to reduce excessive power usage by scaling down the power consumed in the channel so that the SNR of the channel is right on the threshold SNR of the current modulation mode. The power saved in this channel is given to the other channels which only require a small amount of power to boost them up to the next modulation mode. Thus, power allocation increases throughput with limited transmit power while providing the QoS specified by the system.

Chapter 3

Power Allocation

3.1 Power Allocation Basics

MIMO is an effective technique for providing reliable high-data transmission over multipath wireless channels. As the demand for MIMO increases, many researchers have attempted to find techniques to combine with MIMO to further improve the system [6], [7], [8]. One of these techniques is power allocation.

Power (resource) allocation can be used to maximize the number of bits transmitted by varying the transmit power of each channel in an adaptive MIMO system while maintaining the system BER below the target BER. In other words, it maximizes the throughput under power and QoS constraints.

Higher SNR decreases BER. However, if the SNR measured at the receiver can achieve significantly lower BER than the required BER by the system, then some portion of its power is wasted since such a high performance is extravagant. Power allocation is used to give the excess power to a weaker link or other links that require extra power to switch to higher modulation modes to improve the efficiency of the system. In power allocation, transmit power is dynamically allocated in each channel.

Figure 3.1 illustrates a simple application of power allocation in a 2×2 MIMO system. Both measured SNRs have higher power than the minimum required power of 4QAM to achieve the target BER but lower power than

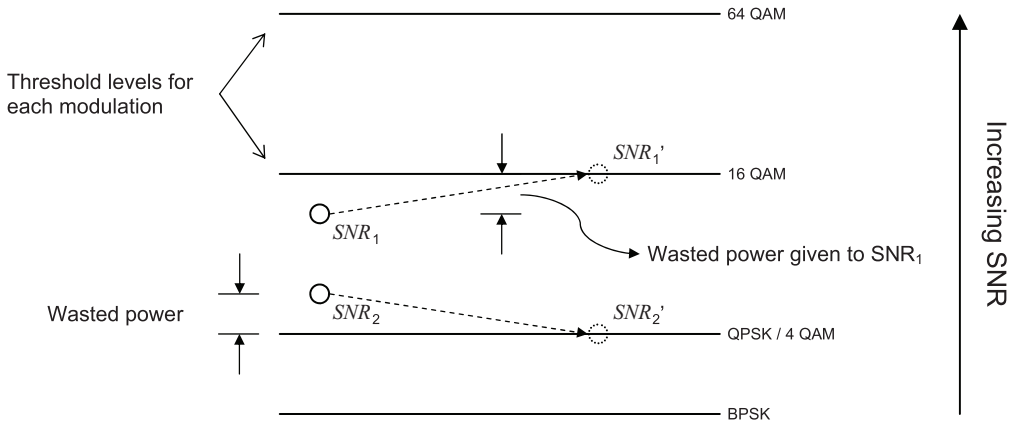


Figure 3.1: Illustration of simple power allocation

the threshold SNR of 16QAM. Thus, both channels use 4QAM to transmit the data, transmitting 4 bits in total. However, giving equal power to both channels is somewhat inefficient since SNR_2 certainly achieves a lower BER than the required BER and an even lower BER is achieved by SNR_1. Thus, precious power is wasted as the SNRs only need enough power just to satisfy the QoS.

Appropriate scaling of transmit power enhances the performance of the system. As depicted in Fig. 3.1, by scaling down SNR_2 to the 4QAM threshold level, we can use the wasted power to boost up SNR_1 to 16QAM. Therefore, we gain an extra 2 bits by power allocation without wasting power or sacrificing BER.

In this thesis, three algorithms are proposed: the fullsearch algorithm (optimal), the fast algorithm, and a modified fast algorithm that is slower but gives a better performance than the fast algorithm.

Signal-to-noise ratios in general are denoted SNR. The particular values for a particular link, say link i, are denoted by SNR_i. Throughout Chapters 3 and 4 these signal-to-noise ratios are generic in the sense that they may refer to SVD systems, MMSE systems, etc.

3.2 Fullsearch Algorithm

The fullsearch algorithm picks an optimal combination that maximizes the throughput from all possible combinations of power scaling factors. For each received SNR, this algorithm computes all the scaling factors required to scale the SNR to each threshold level. For example, assume that the system uses no transmission, BPSK, 4QAM, 16QAM and 64QAM modulations to transmit data symbols. In this situation, four different scaling factors:

i. scaling to the threshold level of BPSK,

ii. scaling to the threshold level of 4QAM,

iii. scaling to the threshold level of 16QAM,

iv. scaling to the threshold level of 64QAM,

are computed for each received SNR. Note that the scaling factor required to scale the SNR to the no transmission mode is zero and does not need to be computed. Thus, this algorithm requires the computation of $m \times (N-1)$ scaling factors where $m = \min(N_t, N_r)$ is the number of channels and N is the number of modulation modes.

After the $m \times (N-1)$ scaling factors are found, the fullsearch algorithm generates the combination matrix, $\mathbf{C} \in \mathbb{R}^{N^m \times m}$ defined as

$$\mathbf{C} = \begin{pmatrix} \text{off} & \text{off} & \cdots & \text{off} \\ \text{BPSK}_1 & \text{off} & \cdots & \text{off} \\ \vdots & \vdots & \cdots & \vdots \\ \text{16QAM}_1 & \text{64QAM}_2 & \cdots & \text{64QAM}_m \\ \text{64QAM}_1 & \text{64QAM}_2 & \cdots & \text{64QAM}_m \end{pmatrix}. \tag{3.1}$$

In (3.1), "BPSK_i", "16QAM_i" etc. represent the power scaling factor of SNR_i required to scale to the corresponding threshold levels. Each row of $\mathbf{C}$ forms a unique combination of the power scaling factors. Since there are m SNRs and N threshold levels, there are N^m possible combinations.

SNR (dB)	Invalid Ratio	SNR (dB)	Invalid Ratio
1	0.8768	11	0.3581
2	0.8439	12	0.2890
3	0.8018	13	0.2334
4	0.7605	14	0.1750
5	0.7136	15	0.1293
6	0.6691	16	0.0842
7	0.6168	17	0.0514
8	0.5581	18	0.0311
9	0.4970	19	0.0132
10	0.4270	20	0.0073

Table 3.1: Invalid combination ratio of the fullsearch algorithm in a 4×4 system.

For each row of $\mathbf{C}$, the fullsearch algorithm computes the number of bits that can be transmitted by that particular combination of scaling factors and checks if the combination is valid. The validity of the combination is checked by comparing the sum of the power scaling factors to m. With the assumption that the average symbol energy is one, the sum of the power scaling factors must not exceed m. For a 4×4 system, when the SNR is very low, e.g., 1dB, about 88% of the 5^4 combinations require more power than the total power. Thus, time is wasted computing invalid combinations. However, as the SNR increases, the invalid ratio reduces. The invalid ratio for SNRs ranging from 1dB to 20dB is given in Table 3.1. Note that a low invalid ratio means that more combinations are valid. Since the fullsearch algorithm compares all the valid combinations, more time is consumed at higher SNRs.

The fullsearch algorithm finds the optimal combination or combinations by checking each possible combination and finding the maximum number of bits provided by all valid combinations. There can be more than one combination which can achieve the maximum possible throughput. By considering all combinations of power scaling factors, the fullsearch algorithm guarantees that the found power scaling factors always maximize the throughput under the power and QoS constraints. Unused power, called the excess power, is

used to improve the system's temporal behavior by giving an extra buffer to the scaled SNRs. Excess power allocation is discussed Chapter 4.

The fullsearch algorithm also considers invalid combinations which increases the computational workload. If the fullsearch algorithm is used in a 4×4 MIMO OFDM system with 64 subcarriers and 5 modulations, this algorithm needs to check 625 combinations for each subcarrier. That is 40,000 combinations in total assuming the same modulation modes are used. Such a huge computational workload makes this algorithm inappropriate for environments where delay is crucial.

3.2.1 Numerical Example of the Fullsearch Algorithm

Consider a 3×3 MIMO system with the instantaneous Rayleigh channel matrix, $\mathbf{H}$, given by

$$\mathbf{H} = \begin{pmatrix} -0.4639 - j0.5286 & +0.8520 + j0.0994 & -1.4270 + j0.6881 \\ -0.2740 - j0.7735 & +0.4400 - j0.2951 & -0.6948 + j0.9975 \\ +0.4872 + j0.8985 & -0.4330 - j0.4220 & -0.7511 + j0.1273 \end{pmatrix}. \quad (3.2)$$

With the SVD technique, the eigenvalues of $\mathbf{HH}^{\dagger}$ and hence the corresponding SNRs for the 3 eigenchannels when the SNR is 10dB are:

$$\begin{aligned} SNR_1 &= \quad 59.3288, \\ SNR_2 &= \quad 21.1796, \\ SNR_3 &= \quad\ 1.1197. \end{aligned}$$

If the target BER of the system is 1×10^{-3}, the threshold SNRs are given in Table 2.1. Therefore, for this system, 5^3 combinations of power scaling factors must be computed. The power scaling factors of each SNR are given in Table 3.2. Thus, the combination matrix, $\mathbf{C} \in \mathbb{R}^{\,5^3 \times 3}$, in this case is shown in (3.3).

SNRs	Scaling Factors				
	Off	BPSK	4QAM	16QAM	64QAM
SNR_1	0	0.0805	0.1610	0.7604	3.0313
SNR_2	0	0.2254	0.4509	2.1300	8.4915
SNR_3	0	4.2643	8.5287	40.2901	160.6198

Table 3.2: Scaling factors to scale the corresponding SNRs to the thresholds.

$$\mathbf{C} = \begin{pmatrix} 0 & 0 & 0 \\ 0.0805 & 0 & 0 \\ \vdots & \vdots & \vdots \\ 0.7604 & 8.4915 & 160.6198 \\ 3.0313 & 8.4915 & 160.6198 \end{pmatrix}. \tag{3.3}$$

The sum of each row must be less than m to be a feasible combination. The sum of a row being greater than m implies that more power is required to satisfy the scaling combination than is available at the transmitter. In this example, SNR_1 and SNR_2 can never be scaled to the 64QAM threshold level and SNR_3 cannot be turned on since the maximum power scaling factor is limited to $m = 3$. Among the $5^3 = 125$ combinations, 109 combinations are invalid. We observe that the fullsearch algorithm can compute many invalid power scaling combinations.

To reduce the computational workload, calculation of the scaling factors of SNR_3 can be omitted. Instead, the scaling factors of the last SNR can be found by subtracting the sum of the power scaling factors of the other SNRs, SNR_1 and SNR_2, from the total. This way of finding the scaling factor of the last SNR reduces the possible combinations to 5^2. However, invalid combinations still exist within these 25 combinations since some of the combinations give a negative value for the scaling factor of the last SNR. Note that one approach to this problem would be to develop highly efficient code to compute the optimal solution. Such code would need to very quickly identify invalid combinations so that only valid combinations are considered. However, it was envisaged that

the complexity of such an approach would still be too high and approximate algorithms were developed instead.

The fullsearch algorithm finds the combination or combinations which maximize the throughput among all possible combinations. In this example, the maximum throughput is 8 bits when both SNR_1 and SNR_2 are scaled to the 16QAM threshold level and SNR_3 is turned off. The resultant (optimal) scaling factors are:

$$\begin{aligned} p_1 &= 0.7604 \quad (16\text{QAM}), \\ p_2 &= 2.1300 \quad (16\text{QAM}), \\ p_3 &= 0 \qquad\quad (\text{Off}), \end{aligned}$$

where p_i is the power scaling factor of SNR_i.

When only adaptive modulation is used in this system, SNR_1 is in 16QAM mode, SNR_2 is in 4QAM mode and SNR_3 is in off mode. Thus, it transmits 6 bits in total. However, when power allocation is also applied, throughput improves from 6 bits to 8 bits. The excess power, $m - \sum p_i$, can be used to improve the system's temporal behavior. The excess power, 0.1096 in this example, is distributed to each SNR to scale up the SNRs further above the threshold levels. The excess power allocation is discussed in Chapter 4.

3.3 Fast Algorithm

Instead of searching through all possible combinations of scaling factors, the fast algorithm attempts to find an optimal power scaling factor for each SNR as the algorithm proceeds in an iterative manner. For each SNR, the algorithm decides whether to scale the SNR up to the next modulation mode or scale it down to the threshold level of the current modulation mode as the algorithm proceeds. For example, if a received SNR is in 16QAM mode, the fast algorithm decides whether to scale the SNR up to 64QAM or scale it down to the threshold level of 16QAM. In other words, the fast algorithm only considers

scaling up to the next modulation mode and scaling down to the threshold level of the current modulation mode.

The fast algorithm pre-computes the following three scaling factors for each SNR:

i. a scaling factor to scale up to the next modulation mode (up scaling factor),

ii. a weighted up scaling factor (decision factor),

iii. a scaling factor to scale down to the threshold level of the current modulation mode (down scaling factor).

Thus, the fast algorithm requires $3 \times m$ scaling factors. Note that an extra factor, the decision factor, is proposed. The reason for this factor is that not all scalings are equal. For example, an up scaling factor of 1.2 in BPSK gives an increase of 1 bit (BPSK $\rightarrow$ 4QAM) whereas a factor of 1.2 in 4QAM gives 2 bits (4QAM $\rightarrow$ 16QAM). Hence, the up scaling should be performed on the SNR in 4QAM first. The decision factor uses this philosophy and gives extra weight to scalings with higher bit increases. In particular, the decision factors are computed as

$$w_i = u_i^{\alpha_i}, \tag{3.4}$$

where u_i is the up scaling factor and α_i is the weighting coefficient. The reason for the power α_i in (3.4) is discussed below. Firstly, we note that the scaling factors are multiplicative so that the up scaling factor, u_i, is multiplied by SNR_i to scale up the SNR to an arbitrary threshold level. If the scaled up SNR, $u_i \times SNR_i$, gives an increase of 2 bits, then we use $\alpha_i = \frac{1}{2}$ and this is the same as $w_i \times w_i \times SNR_i$. If $u_i \times SNR_i$ gives an increase of 3 bits, then we use $\alpha_i = \frac{1}{3}$ and this is equivalent to multiplying w_i^3 by SNR_i. Thus, w_i can be thought of as approximately the power required to gain 1 extra bit. Hence, the weighting coefficient is simply defined as

$$\alpha_i = \frac{1}{\text{No. of bits gained by scaling up } SNR_i}. \tag{3.5}$$

With the modulation modes considered in the thesis, the maximum number of bits obtained by scaling up to the next modulation mode is 2 bits. Therefore, we define the weighting coefficient, α, for SNRs greater than the 4QAM threshold level, to be $\frac{1}{2}$. Hence, the square root of u_i is used as the weighted up scaling factor. i.e., $w_i \times w_i = u_i$.

The fast algorithm starts by scaling up the SNR that corresponds to the minimum decision factor. The flowchart of the fast power allocation algorithm is shown in Fig. 3.2.

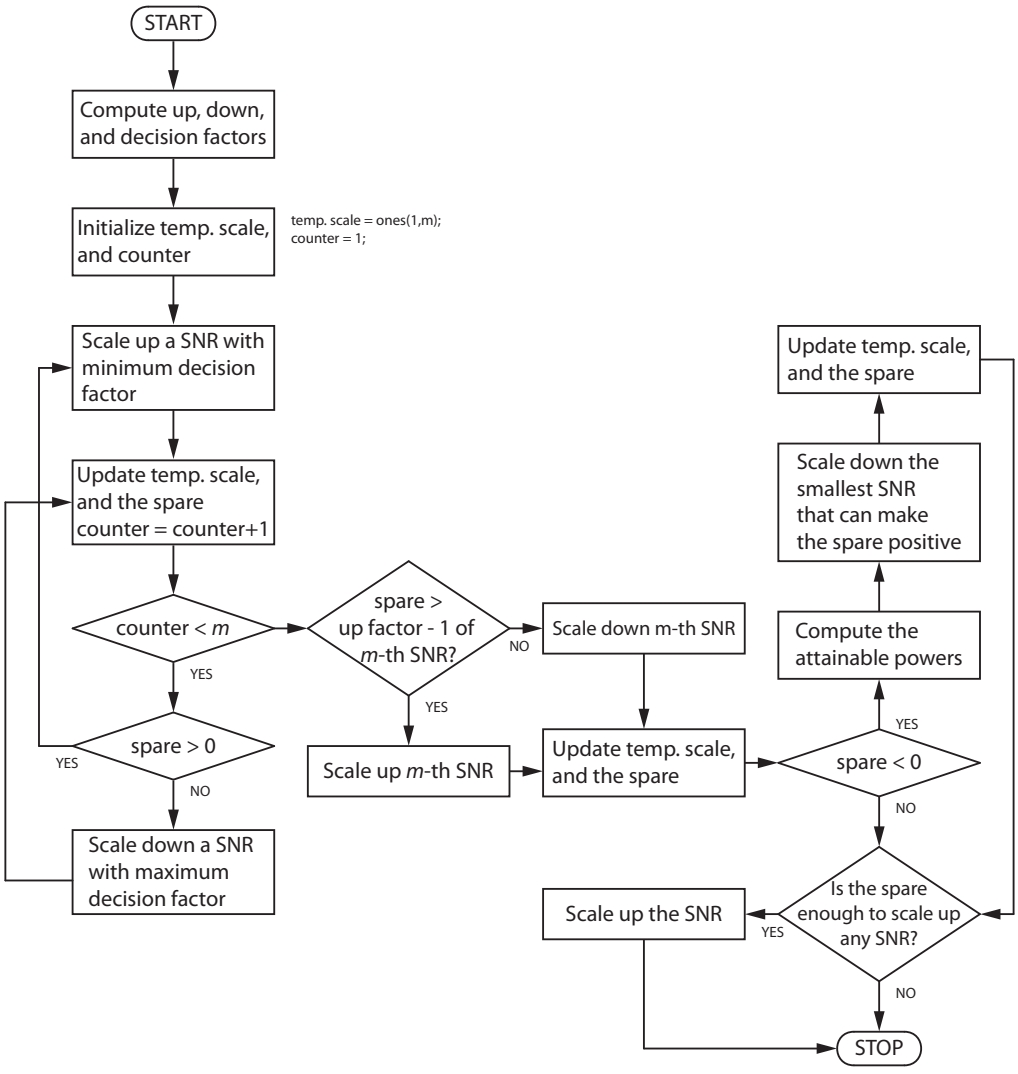


Figure 3.2: Flowchart of the fast power allocation algorithm in a MIMO system.

In the fast algorithm, we introduce a new parameter called the "spare". The spare is a measure of the power that can be used. The spare can be thought as a flag to decide whether to scale up or down the next SNR. After a power scaling factor for an SNR is determined, the fast algorithm computes the spare, which is found by subtracting the sum of temporary scaling factors from m. At the beginning of the algorithm, the temporary scaling factors and the channel multipliers, for each SNR are set to one since none of them are scaled yet. Hence, the spare is zero. A negative value of the spare implies some portion of the power is allocated to a SNR to scale it up. Therefore, some other SNR must be scaled down to keep the spare greater than zero. At the end of the algorithm, the spare must be a positive number, as a negative number means more power is used than the budget.

If the spare after each power allocation is positive, another SNR that corresponds to the second smallest decision factor is scaled up. If the spare is negative, an SNR that corresponds to the maximum decision factor is scaled down. Then, according to the polarity of the spare, this algorithm repeats these scaling up and down steps until it reaches $(m-1)$th SNR.

For the last (mth) SNR, the algorithm checks if the spare is enough to scale up the last SNR. If the spare is enough to scale up the last SNR, it scales it up. Otherwise, the last SNR is scaled down regardless of the polarity of the spare.

After all the temporary power scaling factors are found, the spare is computed again to make sure that the spare is a positive number. Sometimes the spare can be a negative number even though the last SNR is scaled down. If the spare is negative, then the fast algorithm computes attainable powers by scaling down some of the SNRs that were scaled up previously. Hence, the fast algorithm picks an SNR that can make the spare positive. In the case when there exist more than one SNR which can bring the spare up to a positive value, the smallest SNR is picked among the candidate SNRs.

The last part of this algorithm is to find an SNR that can be scaled up by adding the spare to the temporary scaling factors that are less than one. This means that the last part of this algorithm only considers SNRs that are scaled down in previous stages. At this stage, all SNRs are already scaled and usually the spare is not enough to scale up any SNRs.

When all the power scaling factors are found, the temporary scaling factors are locked and the received SNRs are multiplied to scale up or down to maximize the throughput. The spare at the end of the fast algorithm is used to improve the system's temporal behavior as discussed in Chapter 4.

As the fast algorithm picks an optimal power scaling factor for each SNR as it proceeds, the computational load is much lesser than the fullsearch algorithm. However, since the fast algorithm only considers scaling up to the next modulation mode and scaling down to the threshold level of the current modulation mode, the power scaling factor found with the fast algorithm may not be the optimal solution even though it improves the system throughput. Sometimes, the SNRs need to be scaled up to a higher modulation mode or lower modulation mode than an adjacent modulation mode to achieve the optimal throughput. However, the fast algorithm does not consider any scalings beyond the adjacent modes.

3.3.1 Numerical Example of the Fast Algorithm

A 4×4 MIMO system with the received SNRs:

$$\begin{aligned} SNR_1 &= 92.9803, \\ SNR_2 &= 56.4404, \\ SNR_3 &= 20.2005, \\ SNR_4 &= 1.1721, \end{aligned}$$

is considered in this example. The target BER is assumed to be 1×10^{-3} as in the previous examples, and the threshold levels for each modulation mode

	SNR_1	SNR_2	SNR_3	SNR_4
Up scaling factors	1.9342	3.1865	2.2333	4.0738
Decision factors	1.3908	1.7851	1.4944	4.0738
Down scaling factors	0.4852	0.7993	0.4727	0

Table 3.3: The scaling factors computed by the fast algorithm.

are given in Table. 2.1.

The first step of the fast algorithm is to compute the up scaling factors, decision factors and down scaling factors for each received SNR. Thus, the fast algorithm computes 12 scaling factors which are given in Table. 3.3.

Note that SNR_1 and SNR_2 are in 16QAM mode, SNR_3 is in 4QAM and SNR_4 is in no transmission mode. By scaling up SNR_1, SNR_2 and SNR_3 to the next modulation mode, we can gain an extra 2 bits whereas scaling up SNR_4 to BPSK gives only 1 extra bit. Therefore, the decision factor for SNR_4 is the same as the up scaling factor whereas the other decision factors are square-roots of the up scaling factors.

Once these scaling factors are found, the fast algorithm scales up SNR_1 since it has the lowest decision factor. The temporary scaling factor, t_1, of SNR_1 is updated to the corresponding up scaling factor, 1.9342. At this stage, the temporary scaling factors are given in Table. 3.4.

SNR_1	SNR_2	SNR_3	SNR_4
1.9342	1	1	1

Table 3.4: Temporary scaling factors after SNR_1 is scaled up.

Hence, the spare is $m - \sum t_i = 4 - 4.9342 = -0.9342$ where t_i is the temporary scaling factor of SNR_i and m is the number of channels. The spare is negative which means that one SNR must be scaled down. The fast algorithm picks SNR_4 to scale down since SNR_4 has the maximum decision factor. Then, the temporary scaling factors and the spare are updated as in Table. 3.5.

SNR_1	SNR_2	SNR_3	SNR_4	spare
1.9342	1	1	0	0.0658

Table 3.5: Updated temporary scaling factors and the spare after SNR_4 is scaled down.

Note that the spare is now positive. The fast algorithm picks SNR_3 to scale up since the decision factor of SNR_3 is smaller than that of SNR_2. The updated temporary scaling factors and the spare are given in Table. 3.6.

SNR_1	SNR_2	SNR_3	SNR_4	spare
1.9342	1	2.2333	0	-1.1675

Table 3.6: Updated temporary scaling factors and the spare after SNR_3 is scaled up.

Since the spare is negative, SNR_2 must be scaled down. Scaling down SNR_2 the values of the temporary scaling factors and the spare are shown in Table. 3.7.

SNR_1	SNR_2	SNR_3	SNR_4	spare
1.9342	0.7993	2.2333	0	-0.9668

Table 3.7: Updated temporary scaling factors and the spare after SNR_2 is scaled down.

At this stage, all scaling factors have been assigned to the SNRs and the spare is negative. To make the spare positive, the fast algorithm computes the differences between the temporary scaling factors and the down scaling factors to find the candidate SNRs that can make the spare positive. The differences are given in Table. 3.8.

The difference is the power that can be gained by scaling down the SNRs. Hence, the fast algorithm chooses the SNRs that have greater values than the absolute value of the spare. In this example, both SNR_1 and SNR_3 can provide enough power to make the spare positive by scaling down since the attainable power by scaling down SNR_1 and SNR_3 are greater than the absolute value of the spare, 0.9668. Since SNR_3 is lower than SNR_1, SNR_3 is

SNR_1	SNR_2	SNR_3	SNR_4
1.4491	0	1.7605	0

Table 3.8: Attainable power by scaling down SNRs.

chosen to scale down. The temporary scaling factors and the spare are updated as in Table. 3.9. In the final stage, the fast algorithm checks if there is an SNR

SNR_1	SNR_2	SNR_3	SNR_4	spare
1.9342	0.7993	0.4727	0	0.7937

Table 3.9: Final temporary scaling factors and the spare.

that can be scaled up among the scaled down SNRs. In this example, all SNRs are scaled down except SNR_1. Thus, the fast algorithm computes the differences between up scaling factors and the temporary scaling factors, which gives the required power to scale up the SNRs. If there are any channels that have a difference less than the spare, the channel can be scaled up. In this example, the required powers to scale up SNR_2, SNR_3 and SNR_4 are 2.3872, 1.7605 and 4.0738 respectively. All of these require more power than the spare. Therefore, none of them can be scaled up. Thus the final power scaling factors are:

$$\begin{aligned} p_1 &= 1.9342 \quad \text{(64QAM)}, \\ p_2 &= 0.7993 \quad \text{(16QAM)}, \\ p_3 &= 0.4727 \quad \text{(4QAM)}, \\ p_4 &= 0 \qquad\quad \text{(Off)}, \end{aligned}$$

where p_i is the power scaling factor of SNR_i. By applying the power scaling factors the system achieves 12 bits which agrees with the optimal throughput found by the fullsearch algorithm. Note that before the power allocation is done, the throughput was 10 bits.

The same algorithm was also implemented without the use of the decision factors. The up scaling factor was treated as the decision factor in this case. However, the average number of bits transmitted using this method was less

compared to the case when the decision factors are used. Hence, since the use of decision factors has very little overhead their use is preferred.

3.4 Modified Fast Algorithm

The basic idea of the modified fast algorithm is to scale down every received SNR, except the lowest SNR, to the corresponding threshold levels to obtain the maximum spare. The lowest SNR is scaled to no transmission mode, i.e., the lowest SNR is turned off at the beginning of the power allocation process. The spare never goes below zero in the modified fast algorithm. The power gained (the spare) is used to scale up the SNRs in the order of highest priority to lowest priority. The priority of an SNR is decided by two factors:

i. the number of extra bits that can be gained by scaling up the SNR to the next modulation mode,
ii. the power required to scale up the SNR.

The highest priority is given to the SNR that gives the maximum number of extra bits by scaling up to the next modulation mode. The maximum number of bits obtained by scaling up an SNR to the next modulation mode is 2 bits with the modulation modes used throughout this research. If more than one SNR can gain 2 bits by scaling up, higher priority is given to the SNR that requires less power to scale up.

The modified algorithm consists of two main sections as described in Fig. 3.3. In the first section, the SNRs are scaled down and up according to the priority, as long as the spare is positive. When power scaling factors are determined for the higher SNRs, the modified fast algorithm checks if the spare is enough to turn on the lowest channel.

The second part of the modified fast algorithm checks if more than 1 bit can be gained by losing 1 bit. In other words, it checks if enough power is

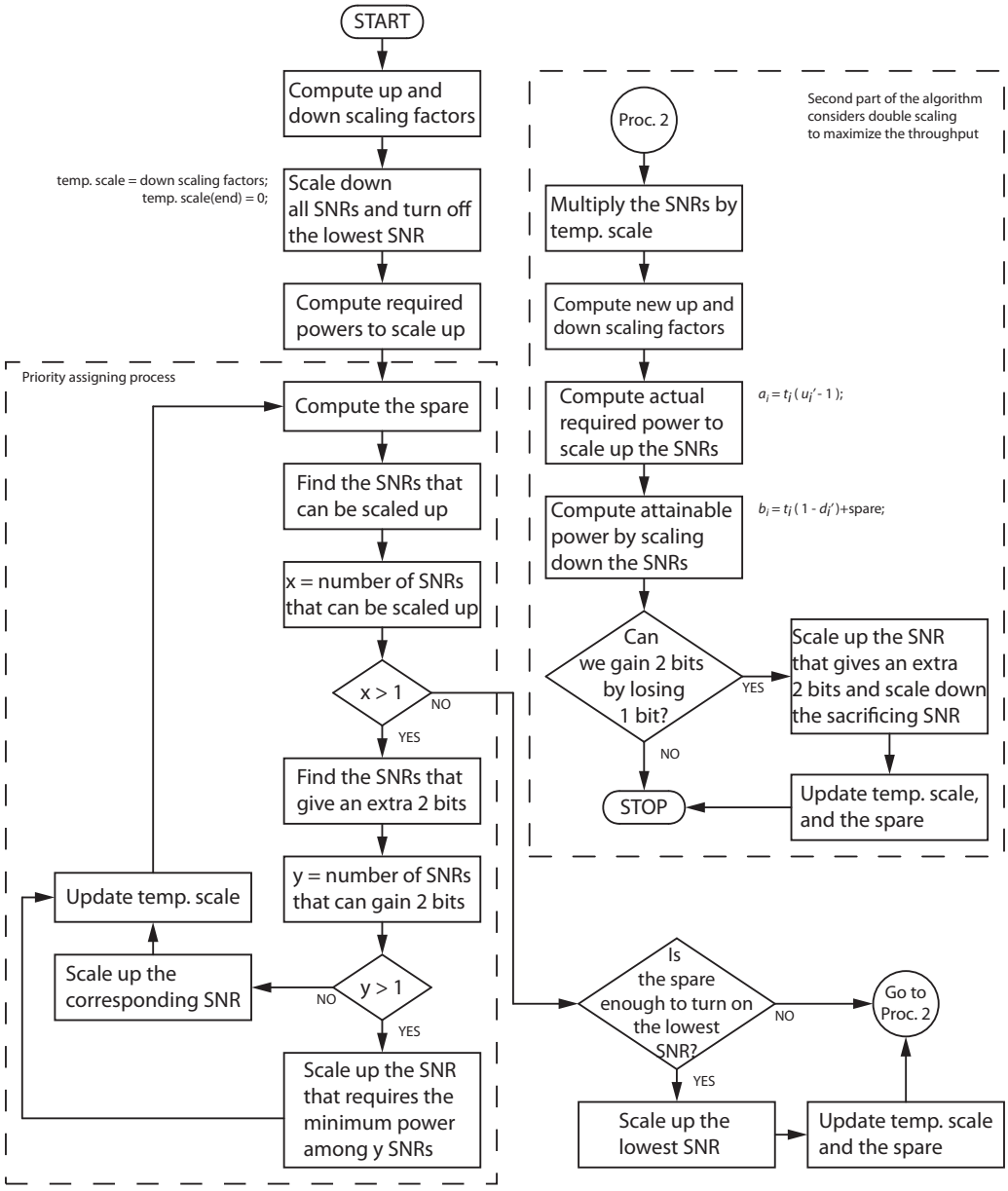


Figure 3.3: Flowchart of the modified fast power allocation algorithm in a MIMO system.

gained to scale up the SNRs that are in 4QAM or higher modulation modes by scaling down the SNRs that are in 4QAM or lower modulation modes.

As for the fast algorithm, at the beginning of the modified fast algorithm the up scaling factors and the down scaling factors are computed. Note that the decision factors are not required for the modified fast algorithm. Thus, the modified fast algorithm requires $2 \times m$ scaling factors to begin with. When these scaling factors are found, it turns off the last SNR and sets the temporary scaling factors, the channel multipliers, of the SNRs to down scaling factors. Hence, the spare is positive at the start of the algorithm.

To assign priorities to each SNR, the algorithm finds out the current modulation mode of the channel and the required power to scale the SNR up to the next modulation mode. The required power to scale up the SNRs is not the same as the up scaling factor as the SNRs are already scaled down. The required power can be found simply by subtracting the temporary scaling factors from the up scaling factors. Thus, this algorithm gives the highest priority to the SNRs that can gain an extra 2 bits. If there is more than one candidate, the SNR that requires the minimum power to scale up is chosen. Then, the temporary scaling factor of the SNR is updated to the corresponding up scaling factor. The spare is computed after each scaling and is found in the same way as in the fast algorithm, i.e., spare $= m - \sum$ temp. scale.

After an SNR is scaled up, the algorithm searches for other SNRs that can be scaled up. Even if an SNR gives an extra 2 bits, the SNR cannot be scaled up if it requires more power than the spare. Hence, the modified fast algorithm assigns a new priority among the feasible scalings. The modified algorithm repeats this scaling up process until the spare goes below the minimum of the required powers to scale up any SNR. After each scaling of an SNR, the corresponding temporary scale of the SNR and the spare are updated.

Once the spare falls below the minimum required power, it checks if the lowest channel can be turned on. In other words, it checks if the spare is

enough to scale the lowest SNR to the BPSK threshold level. If the spare is insufficient to turn on the bottom channel, it remains turned off. Then, the algorithm moves on to the second section of the algorithm.

In the second section, the modified fast algorithm computes the new SNRs; SNRs multiplied by the corresponding temporary scaling factors. In addition, new up scaling factors and new down scaling factors are computed; i.e. up and down scaling factors of the new SNRs. Thus, the modified fast algorithm requires the computation of $4 \times m$ scaling factors in total.

The modified fast algorithm computes attainable powers by scaling down the SNRs that are in 4QAM or lower modulation mode. Scaling down these SNRs loses 1 bit. The attainable power is then compared with the new up scaling factors of the SNRs that are in 4QAM or higher modulation modes to check if we can gain 2 bits by losing 1 bit.

The modified fast algorithm has higher computational workload than the fast algorithm since it computes more scaling factors and more comparisons are performed than in the fast algorithm. However, the modified fast algorithm provides a slightly better performance than the fast algorithm.

3.4.1 Numerical Example of the Modified Fast Algorithm

A 4×4 MIMO system with the received SNRs:

$$\begin{aligned} SNR_1 &= \ 87.2591, \\ SNR_2 &= \ 30.5335, \\ SNR_3 &= \ 13.0965, \\ SNR_4 &= \ \ 1.0590, \end{aligned}$$

is considered in this example. The target BER is assumed to be 1×10^{-3}, as in the previous examples, and the threshold levels for each modulation mode

	SNR_1	SNR_2	SNR_3	SNR_4
Up scaling factors	2.0611	1.4775	3.4446	4.5086
Down scaling factors	0.5170	0.3128	0.7292	0

Table 3.10: The scaling factors computed by the modified fast algorithm

are given in Table. 2.1. Note that SNR_1 is in 16QAM mode (4bits), SNR_2 and SNR_3 are in 4QAM mode (2bits), and SNR_4 is in no transmission mode.

The first step of the modified algorithm is to compute the up scaling factors and down scaling factors. The scaling factors to scale up and down are given in Table 3.10.

The next step of this algorithm is to turn off the lowest channel and to scale down the other SNRs. Therefore, in this example, the temporary scaling factors, the channel multipliers, are the same as the down scaling factors and the spare is $4 - \sum t_i = 2.4411$ where t_i is the temporary scaling factor of SNR_i. The temporary scaling factors at the beginning of the algorithm are given Table. 3.11.

SNR_1	SNR_2	SNR_3	SNR_4	spare
0.5170	0.3128	0.7292	0	2.4411

Table 3.11: Initial temporary scaling factors and the spare.

Then, the modified algorithm assigns a priority to each SNR. Assigning priorities takes 3 steps. First, the modified algorithm checks which SNRs are feasible to be scaled up. If the required power of a SNR is less than the spare, the SNR can be scaled up. The required powers to scale up SNRs are simply up scaling factors subtracted by the temporary scaling factors and are given Table. 3.12.

SNR_1	SNR_2	SNR_3	SNR_4	spare
1.5441	1.1647	2.7154	4.5086	2.4411

Table 3.12: Required power to scale up the SNRs.

The required power to scale up SNR_1 or SNR_2 is less than the spare. Hence, these SNRs can be scaled up.

The second step of assigning priorities is to check how many extra bits the SNRs provide by scaling up. Scaling up SNR_1 to 64QAM gives an extra 2 bits and scaling up SNR_2 to 16QAM also gives an extra 2 bits.

The last step of assigning priorities is to check which SNR requires the minimum power to scale up to the next modulation modes. Since SNR_2 requires the minimum power, the first priority is given to SNR_2. In general, the priorities are given to SNRs, which passed through step 1 and step 2, in the order of minimum required power to the maximum. Thus, the second priority is given to SNR_1.

Since the first priority is given to SNR_2, it gets scaled up. Hence, the temporary scaling factors and the spare are updated as in Table. 3.13. The temporary scaling factor of SNR_2 is updated to its up scaling factor given in Table 3.10 and temporary scaling factors of other SNRs remain the same.

SNR_1	SNR_2	SNR_3	SNR_4	spare
0.5170	1.4775	0.7292	0	1.2763

Table 3.13: Updated temporary scaling factors and the spare.

Now, the modified fast algorithm checks if the second priority SNR can be scaled up. The required power to scale up the second priority SNR, SNR_1, is 1.5441 which is greater than the spare. Therefore, SNR_1 cannot be scaled up.

Then, it checks if the spare is enough to turn on the lowest SNR, SNR_4. The required power to scale up SNR_4 is 4.5086 which also exceeds the spare. Hence, the lowest SNR, SNR_4, cannot be scaled up to BPSK mode either. Thus, the modified fast algorithm ends the first section of the algorithm and moves on to the second section. At the end of the first section of the modified fast algorithm, the throughput has improved to 10 bits from 8 bits since SNR_2 is scaled up to 16QAM and the other SNRs stayed in the same modulation mode.

In the second section of the algorithm, new up scaling factors and down scaling factors of the scaled SNRs are computed. In this example, SNR_1 and SNR_2 are scaled to the threshold level of 16QAM and SNR_3 is scaled down to 4QAM while SNR_4 remained in no transmission. New up scaling factors and new down scaling factors are given in Table. 3.14.

	SNR_1	SNR_2	SNR_3	SNR_4
New up scaling factors	3.9866	3.9866	4.7241	∞
New down scaling factors	0.2117	0.2117	0.5000	0

Table 3.14: New up and down scaling factors computed in the second section of the modified fast algorithm.

Note that the new up scaling factor of SNR_4 is ∞ since the temporary scaling factor of SNR_4 was 0. These scaling factors are not the same as the actual scaling factors that are required to scale up or down the SNRs. Computation of the new up and down scaling factors assumes that the SNRs are right on a threshold level. Since the received SNRs are multiplied by the temporary scaling factors, t_i, to scale the SNRs to the threshold levels, the actual up and down scaling factors are obtained by the multiplying the above scaling factors by the temporary scaling factors. These new scaling factors are used to compute the actual required power to scale up and the attainable power by scaling down the SNRs. The actual required power to scale up SNR_i is computed as

$$a_i = t_i(u_i' - 1), \tag{3.6}$$

where t_i, and u_i' are the temporary scaling factor and new up scaling factor of SNR_i respectively. To boost a SNR by two modulation levels, the overall power scaling factor is uu' where u is the up scaling factor. Hence, the actual required powers to scale up the SNRs are shown in Table. 3.15.

For the SNRs that have infinite power as the new up scaling factor, the actual required power to scale up is the same as the previous up scaling factor. Thus, the actual required power to scale up SNR_4 is 4.5086 as before.

SNR_1	SNR_2	SNR_3	SNR_4
1.5441	4.4126	271.5481	4.5086

Table 3.15: Actual power required to scale up SNRs.

In addition, the attainable powers by scaling down SNRs must be computed to check if we can gain an extra 2 bits by losing 1 bit. Similarly, the attainable power by scaling down SNR_i is computed as

$$b_i = t_i(1 - d_i') + \text{spare}, \tag{3.7}$$

and is given in Table. 3.16.

SNR_1	SNR_2	SNR_3	SNR_4
1.6839	2.4411	1.6409	1.2763

Table 3.16: Attainable power by scaling down SNRs.

Among the actual powers required to scale up the SNRs, the modified fast algorithm only considers the SNRs that are in 4QAM or higher modulation modes. Moreover, the attainable powers corresponding to SNRs that are in 4QAM or lower modulation modes are considered. In other words, scaling up SNR_1, SNR_2 or SNR_3 by scaling down SNR_3 or SNR_4 are considered. However, in this example SNR_4 is already in off mode.

Attainable power computation shows scaling down SNR_3 gives 1.6409 which is enough to scale up SNR_1; the required power to scale up SNR_1 is 1.5441. Then, the fast modified algorithm chooses to scale up SNR_1 to 64QAM and scale down SNR_3 to BPSK. The power scaling factor of SNR_1 is updated to $p_1 = t_1 \times u_1'$ and the final power scaling factors are given below.

$$\begin{aligned} p_1 &= 2.0611 \quad \text{(64QAM)}, \\ p_2 &= 1.4775 \quad \text{(16QAM)}, \\ p_3 &= 0.3646 \quad \text{(BPSK)}, \\ p_4 &= 0 \qquad\quad \text{(Off)}. \end{aligned}$$

Note that SNR_3 is scaled down twice. Applying the power allocation found by the modified fast algorithm achieves 11 bits which agrees with the optimal number of bits found by the fullsearch algorithm. Note that the system was transmitting 8 bits before the power allocation is applied. The throughput was increased to 10 bits after the first section of the modified fast algorithm. The second section of the modified fast algorithm gives an increase of 1 bit.

3.5 Simulation Results

This section compares the optimal throughput found by the fullsearch algorithm with the throughput found by the fast algorithm and the modified fast algorithm. The instantaneous channel matrices are obtained by (2.3), (2.6), and (2.7) for the Rayleigh, the Rician, and the correlated channels respectively. Each sample point is obtained by simulating 10,000 times and taking the mean of the 10,000 values. Note that the first set of results, Figs. 3.4 - 3.21, all assume i.i.d Rayleigh fading.

In a 2×2 SVD system, both the fast algorithm and the modified algorithm show good agreement with the optimal throughput computed by the fullsearch algorithm as shown in Fig. 3.4. On average, the result of the fast algorithm shows 98.3% agreement with the fullsearch algorithm (optimal). The maximum number of bits transmitted by the fullsearch and the fast algorithm differed by at most 2 bits in the simulations.

The modified fast algorithm shows a slightly better performance than the fast algorithm as expected. The modified algorithm shows about 99.9% agreement with the optimal throughput on average. In the low SNR region, SNR $<$ 10dB, it shows 100% agreement with the fullsearch algorithm. The instantaneous throughput difference between the optimal number of bits transmitted and the modified fast algorithm can be as large as 2 bits as for the fast algorithm. In 2×2 SVD system, the fast algorithm and the modified fast

algorithm do not show a noticeable difference in performance.

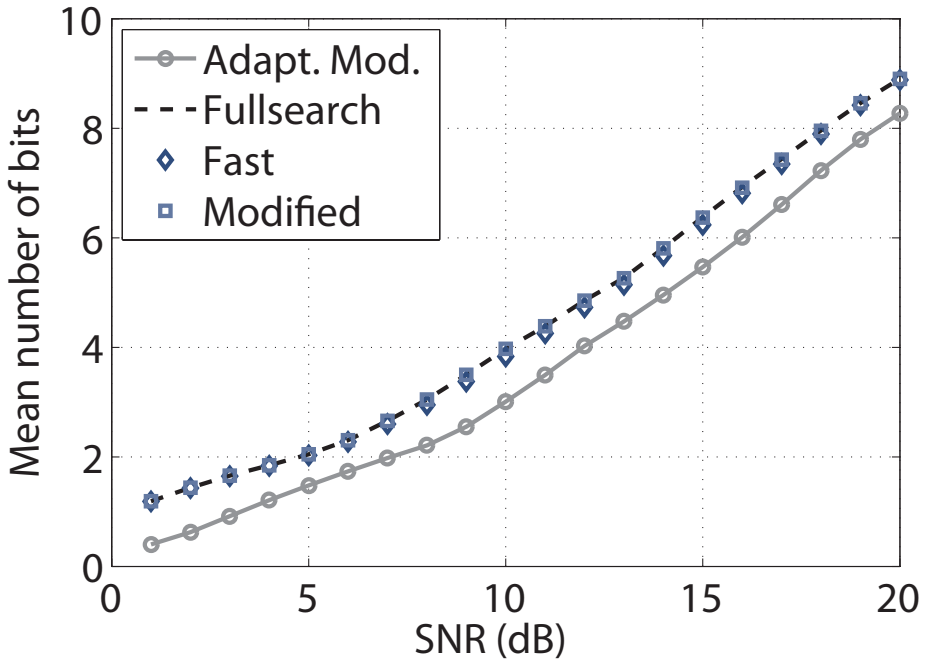


Figure 3.4: Number of bits transmitted in a 2×2 SVD system with target BER $= 1 \times 10^{-3}$.

To compare performance of the MMSE system with the SVD system, the same channel matrices are used to plot Fig. 3.5. Also in the MMSE system, the fast algorithm and the modified fast algorithm show on average, 97.7% and 99.8% agreement with the optimal result respectively. However, the mean number of bits transmitted are slightly lower compared to the SVD system, especially when the SNR is around 15dB. The difference in the mean number of bits transmitted between SVD and MMSE is about 1.4 bits at 17dB. As shown in Fig. 3.8, the difference between SVD and MMSE gets smaller after 17dB.

The performance of a 4×4 SVD system and a 4×4 MMSE system are given in Fig. 3.6 and Fig. 3.7 respectively. In both figures, the performance of the fast algorithm and modified fast algorithm gets better as the SNR increases. In the 4×4 SVD system, the fast algorithm shows about 95.7% agreement with

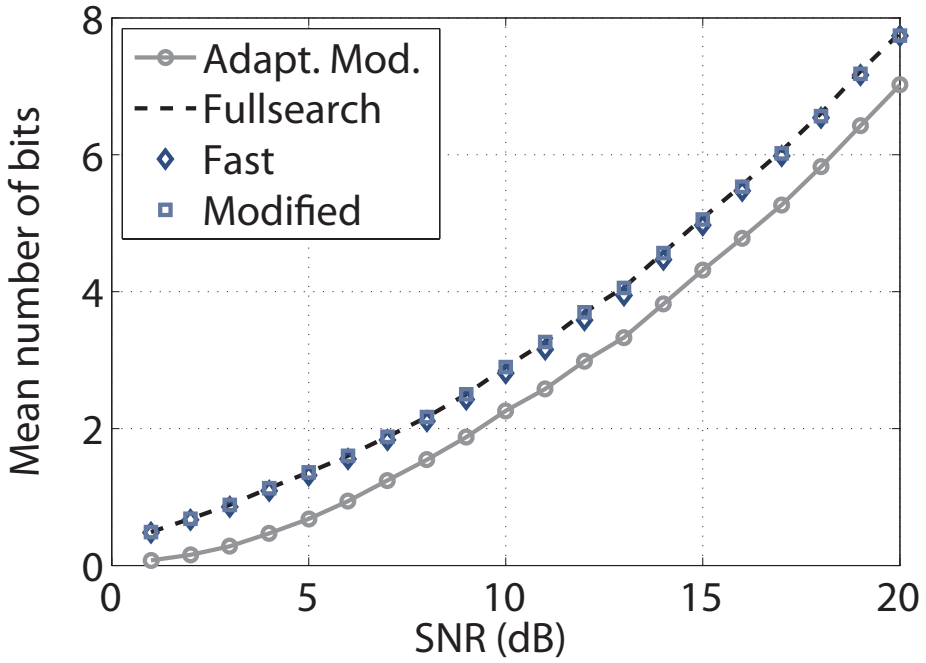


Figure 3.5: Number of bits transmitted in a 2×2 MMSE system with target BER $= 1 \times 10^{-3}$.

the optimal result in the low SNR region and it increases up to 99.8% in the high SNR region. The modified fast algorithm shows about 96.9% agreement in the low SNR region and about 99.9% agreement in the high SNR region.

For the 4×4 MMSE system, the fast algorithm shows about 87.4% agreement in the low SNR region and it increases up to 99.4% as the SNR increases. However, ironically the instantaneous throughput difference between the number of bits transmitted by the fullsearch and the fast algorithm increases as the SNR increases. The maximum throughput differences at SNRs less than 5dB is 1 bit, 2 bits around 10dB, and at SNRs higher than 15dB, the maximum instantaneous throughput difference increases to 4 bits. Thus, as the SNR increases the probability of getting the optimal power scaling factors increases but the possible instantaneous throughput difference also increases. This is simply due to the fact that at low SNR only small rates are possible so the improvements offered by the optimal method are also small.

For the modified fast algorithm also, the maximum instantaneous throughput difference to the optimal throughput increases as the SNR increases. Results from the modified algorithm show that the maximum instantaneous throughput difference for SNRs lower than 15dB is 1 bit and 2 bits for SNRs higher than 15dB. In the low SNR region, the modified fast algorithm shows about 98.3% agreement with the optimal result and about 99.5% agreement at 20dB.

Hence, in a 4×4 MIMO system, the overall performance of the fast algorithm and the modified algorithm are similar. However, in this sequence of simulations, the power scaling factors determined by the fast algorithm can make the instantaneous throughput 4 bits less than the optimal result whereas the modified fast algorithm show only 2 bits difference.

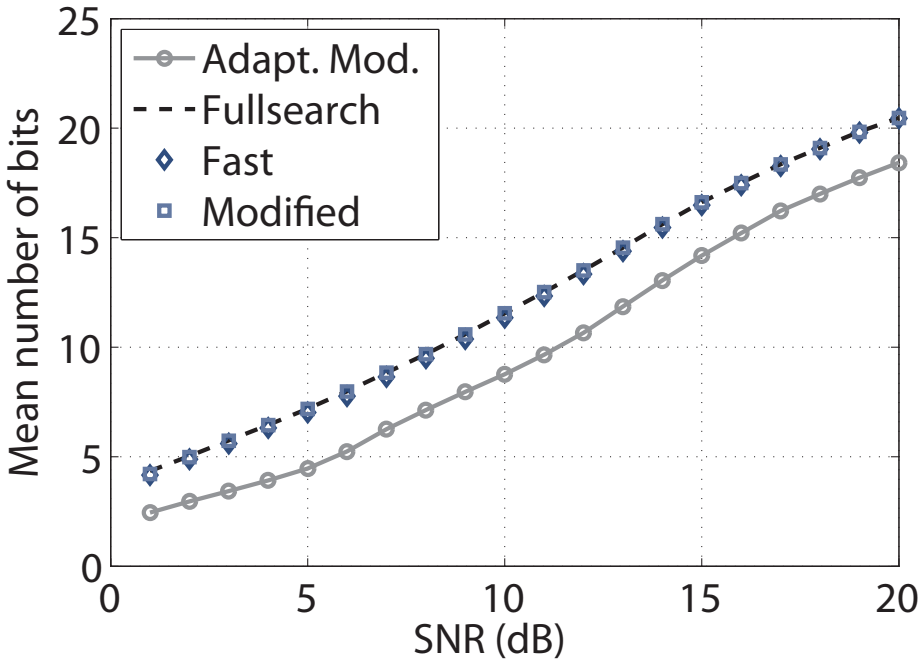


Figure 3.6: Number of bits transmitted in a 4×4 SVD system with target BER $= 1 \times 10^{-3}$.

As shown in Figs. 3.4 - 3.7, both the fast algorithm and the modified fast algorithm show good performance; both algorithms achieve over 95% agreement

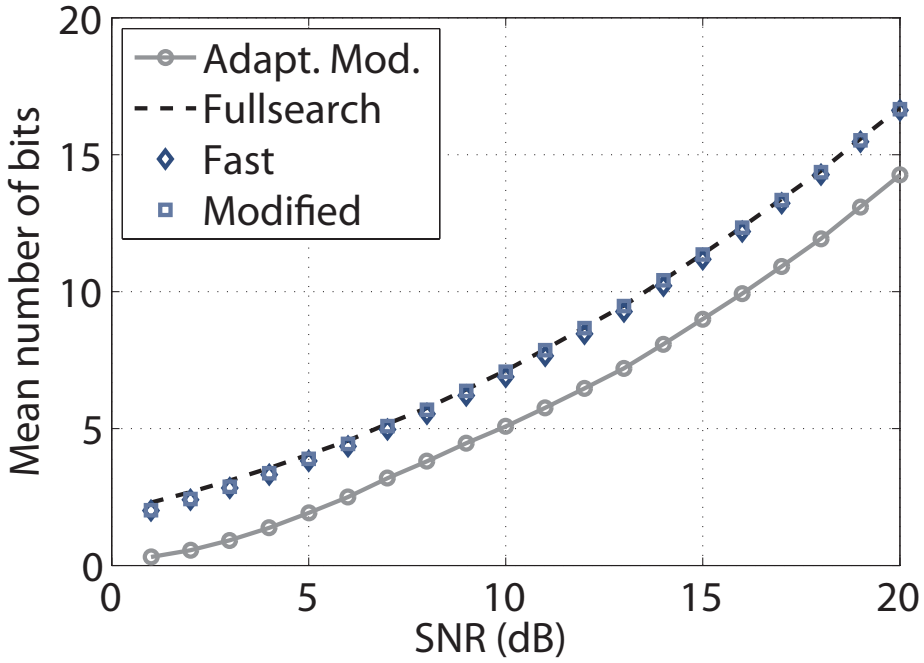


Figure 3.7: Number of bits transmitted in a 4×4 MMSE system with target BER $= 1 \times 10^{-3}$.

on average with the optimal power scaling factors computed by the fullsearch algorithm. Hence, the quickest algorithm to find the power scaling factors is preferred.

The mean CPU time taken to find the power scaling factors for the Rayleigh channel is given in Tables 3.17 - 3.19 for different system sizes. Note that in the low SNR region, the fullsearch algorithm is similar to the modified fast algorithm in a 2×2 system. However, the time taken by the fullsearch algorithm increases enormously as the system size gets bigger. Moreover, the time taken by the fullsearch algorithm also increases as the SNR increases whereas the fast and the modified fast algorithm do not show any relationship between the processing time and the SNR. This is due to the decreased invalid combination ratio at higher SNRs as shown in Table. 3.1. Hence, as the SNR increases, the fullsearch algorithm must compare more valid combinations which consequently increases the processing time. Thus, one would pick the fast algorithm

SNR (dB)	Mean CPU time taken in 2×2 system (ms)		
	Fullsearch Algorithm	Fast Algorithm	Modified Algorithm
1	0.6250	0.5000	0.7188
2	0.5781	0.6250	0.5469
3	0.5469	0.4844	0.5469
4	0.5469	0.5000	0.5469
5	0.5156	0.4844	0.5625
6	0.5781	0.5000	0.5625
7	0.5781	0.4688	0.5469
8	0.6094	0.5000	0.5625
9	0.5938	0.5000	0.5625
10	0.6250	0.5000	0.5625
11	0.6719	0.4844	0.5469
12	0.6875	0.5000	0.5625
13	0.7031	0.5000	0.5625
14	0.7188	0.5000	0.5625
15	0.7500	0.4688	0.5625
16	0.7656	0.4844	0.5625
17	0.7969	0.4844	0.5469
18	0.8438	0.4844	0.5313
19	0.8281	0.5156	0.5625
20	0.8594	0.4844	0.5625
Mean	0.6711	0.4985	0.5641

Table 3.17: Mean CPU time taken (ms) to compute the power scaling factors in a 2×2 system.

to be the best algorithm for the power allocation.

Note that the fullsearch algorithm mean run-time increases nearly 10-fold from 2×2 to 4×4 whereas the fast algorithm only increases by about 20%. In a 6×6 system, the mean CPU time taken by the fullsearch algorithm is about 2300 times longer than the fast algorithm at 20dB.

The SVD systems are superior to the MMSE systems, especially for larger system sizes as shown in Figs. 3.4 - 3.7. In addition, more transmit and receive antennas induce higher throughput. Comparisons between SVD systems and MMSE systems using the fullsearch algorithm are given in Fig. 3.8. Note that the notation (N_r, N_t) is used to denote a $N_r \times N_t$ system size in Fig 3.8.

Note that the performance of the 4×2 MMSE system is almost as good

SNR (dB)	Mean CPU time taken in 4×4 system (ms)		
	Fullsearch Algorithm	Fast Algorithm	Modified Algorithm
1	2.6250	0.6719	0.8125
2	2.7969	0.5938	0.7344
3	2.8906	0.6563	0.7031
4	2.9688	0.5938	0.7031
5	3.1875	0.5781	0.7031
6	3.4531	0.5781	0.7969
7	3.7344	0.5781	0.7031
8	4.2969	0.5781	0.7031
9	4.9063	0.5781	0.7031
10	5.5469	0.5938	0.6875
11	6.4688	0.5781	0.7031
12	7.4219	0.5781	0.7188
13	8.3594	0.5781	0.6875
14	9.5938	0.5781	0.7031
15	10.9844	0.5938	0.7031
16	12.1563	0.5781	0.6875
17	13.3281	0.5938	0.6875
18	14.6094	0.5781	0.6875
19	15.7500	0.5781	0.6719
20	17.0469	0.5781	0.6719
Mean	7.6063	0.5906	0.7086

Table 3.18: Mean CPU time taken (ms) to compute the power scaling factors in a 4×4 system.

SNR (dB)	Mean CPU time taken in 6×6 system (ms)		
	Fullsearch Algorithm	Fast Algorithm	Modified Algorithm
1	55.5	0.7656	0.9219
2	58.6	0.7813	0.8750
3	61.6	0.6875	0.9688
4	68.1	0.6875	0.8594
5	75.2	0.6875	0.8438
6	90.3	0.6719	0.8594
7	110.8	0.7031	0.8438
8	154.8	0.6875	0.8594
9	249.4	0.6875	0.8594
10	347.3	0.6875	0.8594
11	509.8	0.6875	0.8594
12	943.9	0.6875	0.8438
13	957.0	0.6875	0.8594
14	1575.9	0.6875	0.8438
15	2086.3	0.6875	0.8438
16	2801.6	0.6875	0.8438
17	4127.8	0.6875	0.8438
18	5082.3	0.6875	0.8438
19	6183.1	0.6875	0.8281
20	6720.2	0.6875	0.8125
Mean	1613.0	0.6961	0.8586

Table 3.19: Mean CPU time taken (ms) to compute the power scaling factors in a 6×6 system.

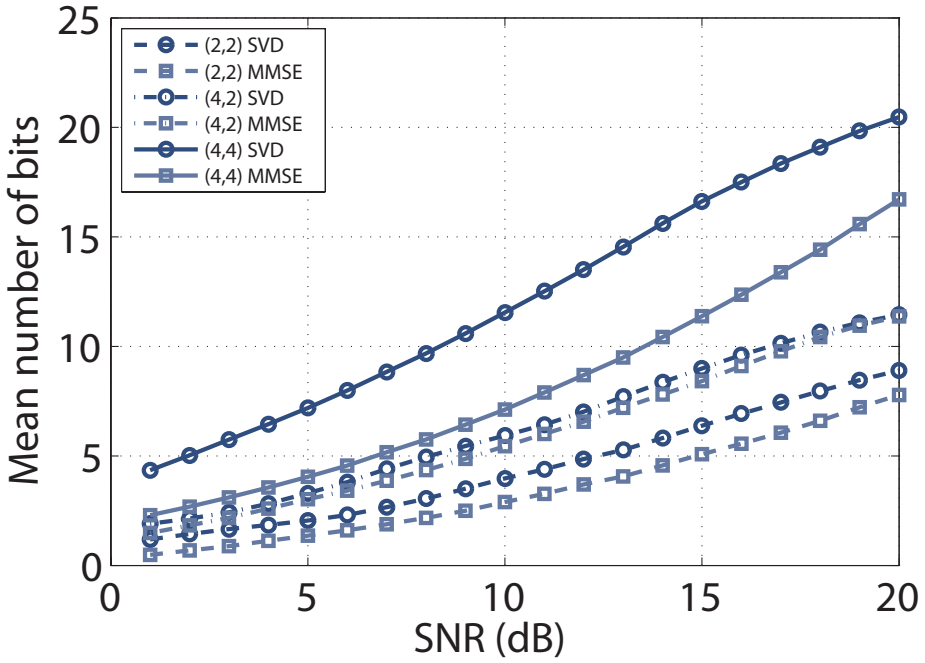


Figure 3.8: Comparisons between SVD systems and MMSE systems for various system sizes. Target BER $= 1 \times 10^{-3}$.

as the SVD system. Even though the overall MMSE performance is slightly inferior to the SVD system, the difference between the mean throughput of the SVD and the MMSE system is less than 0.5 bits. As we can observe in Figs. 3.12 - 3.14, the mean received SNRs of the 4×2 MMSE system are only about 2dB less than the mean received SNRs of the corresponding SVD system. An asymmetric MIMO system, $N_r \times N_t, N_r > N_t$, offers an improved performance for MMSE relative to SVD compared to a $N_t \times N_t$ symmetric MIMO system.

Figures 3.9 - 3.20 show the mean received SNRs in 2×2, 4×2, 4×4, and 8×8 systems for SNRs at 1dB, 10dB, and 20dB. The results support the well known conclusion that the throughput of SVD systems are higher than MMSE systems.

Note that as the system size increases, the gap between the highest SNR in the SVD system and the highest SNR in the MMSE system also increases.

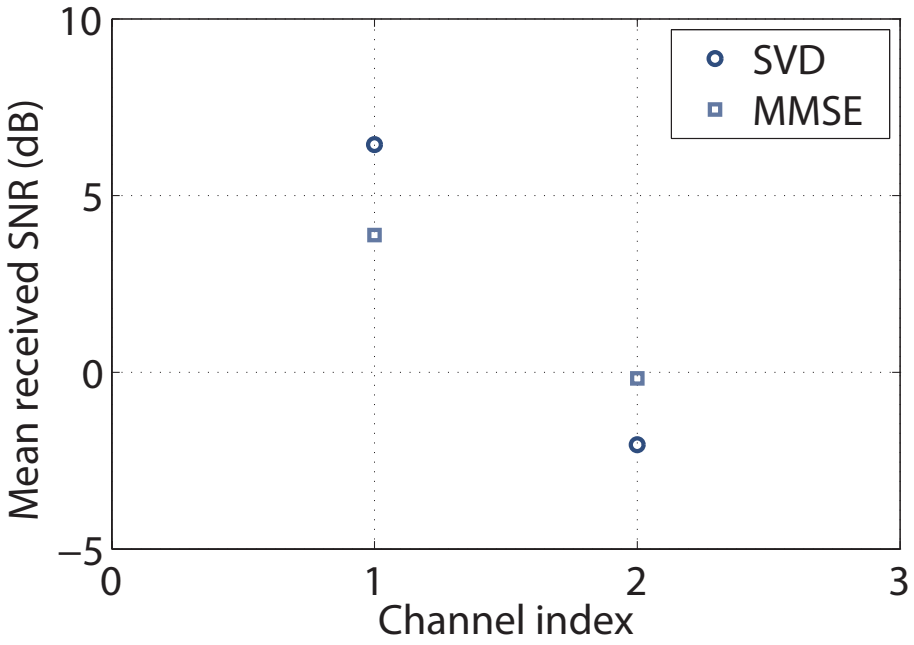


Figure 3.9: Mean received SNRs in a 2×2 system at 1dB.

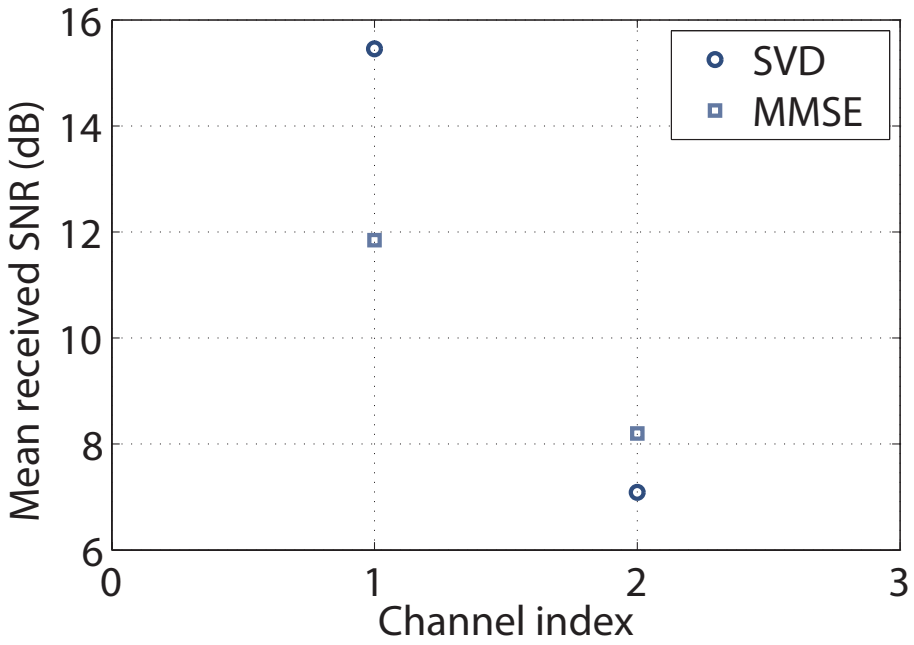


Figure 3.10: Mean received SNRs in a 2×2 system at 10dB.

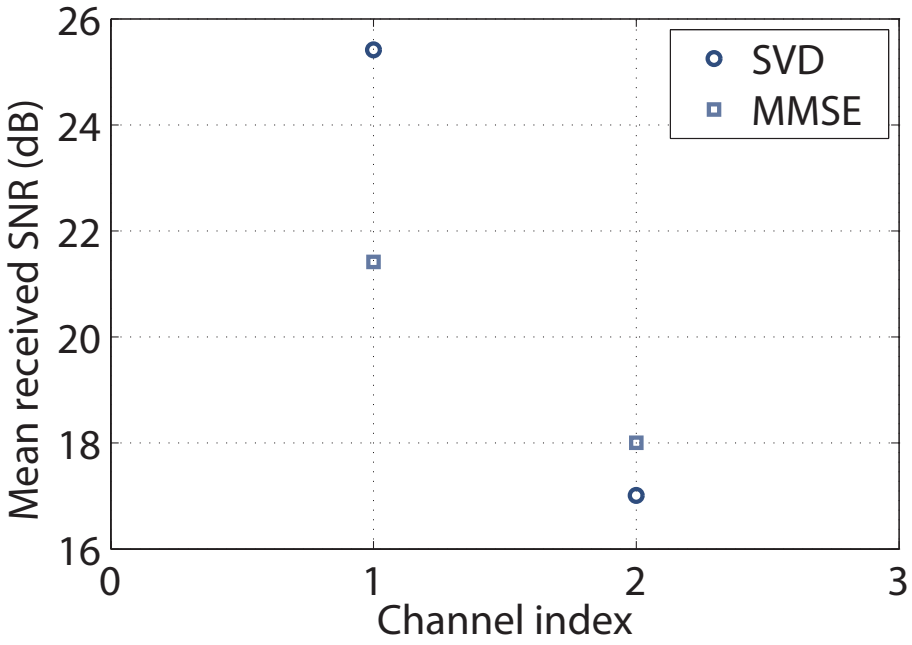


Figure 3.11: Mean received SNRs in a 2×2 system at 20dB.

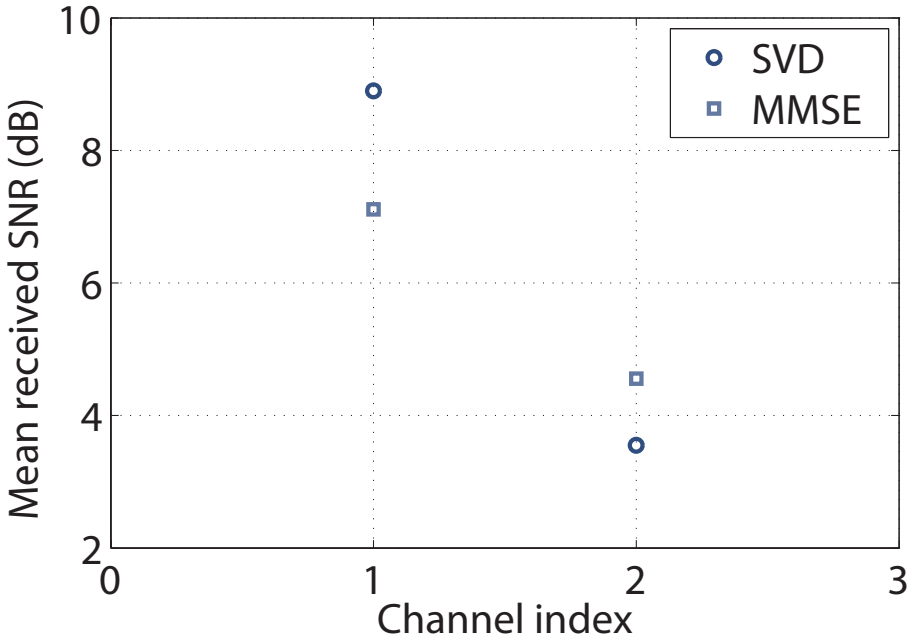


Figure 3.12: Mean received SNRs in a 4×2 system at 1dB.

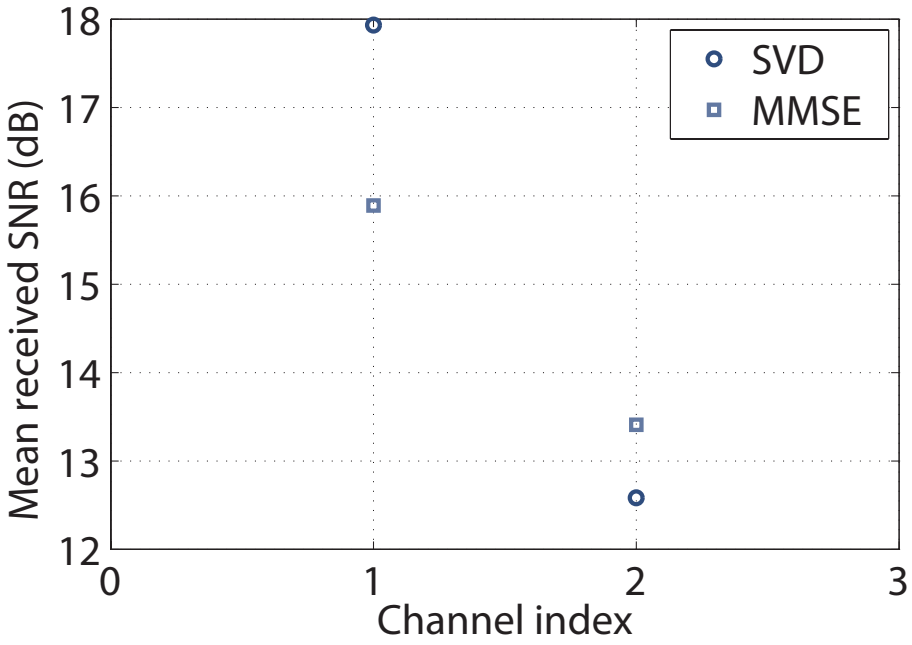


Figure 3.13: Mean received SNRs in a 4×2 system at 10dB.

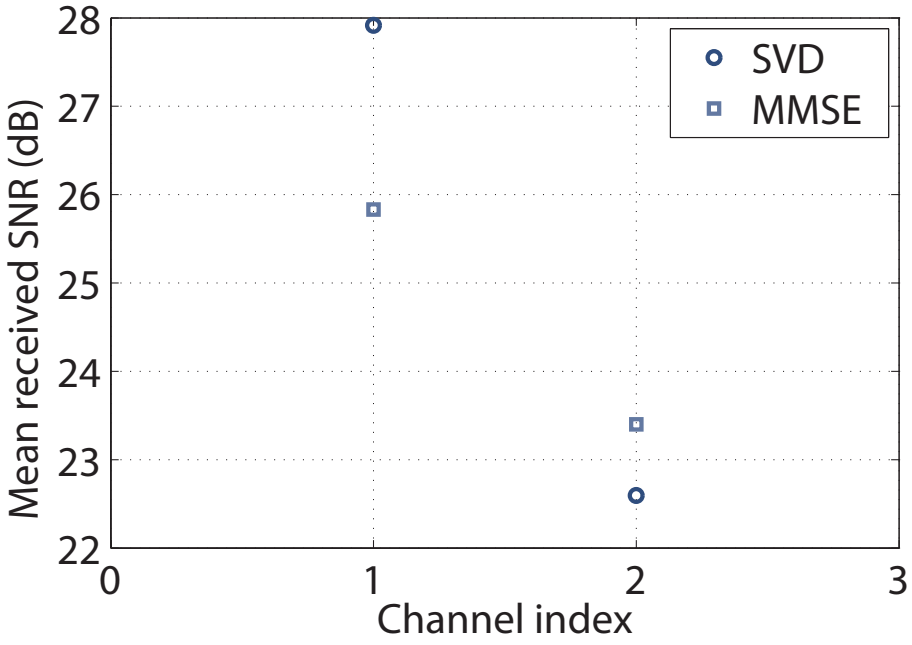


Figure 3.14: Mean received SNRs in a 4×2 system at 20dB.

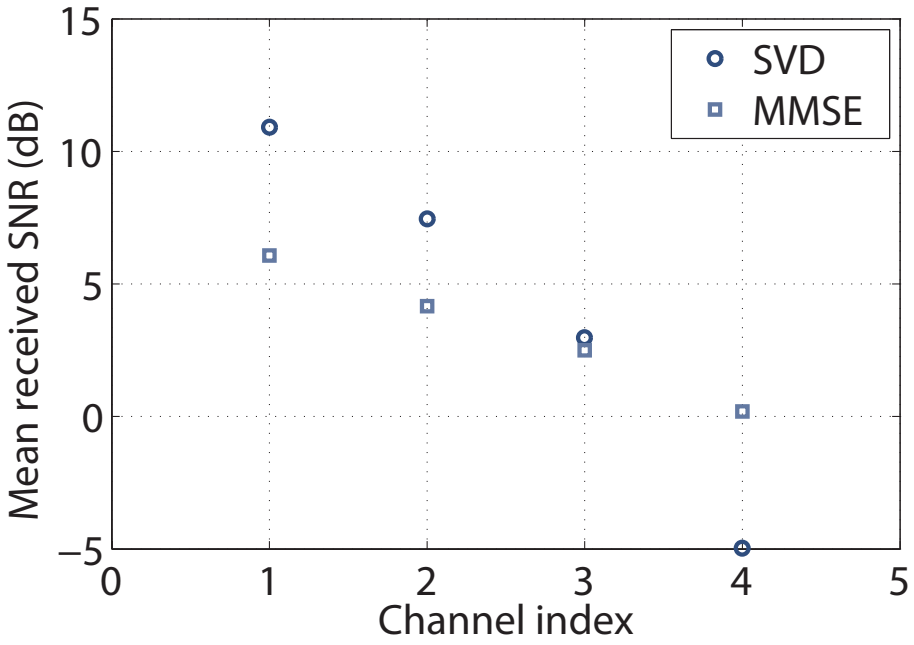


Figure 3.15: Mean received SNRs in a 4×4 system at 1dB.

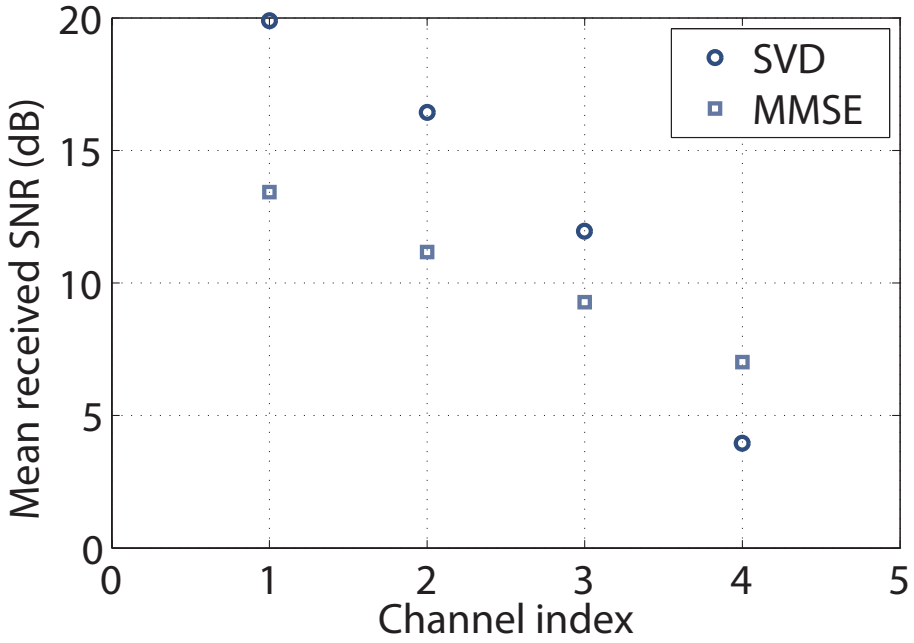


Figure 3.16: Mean received SNRs in a 4×4 system at 10dB.

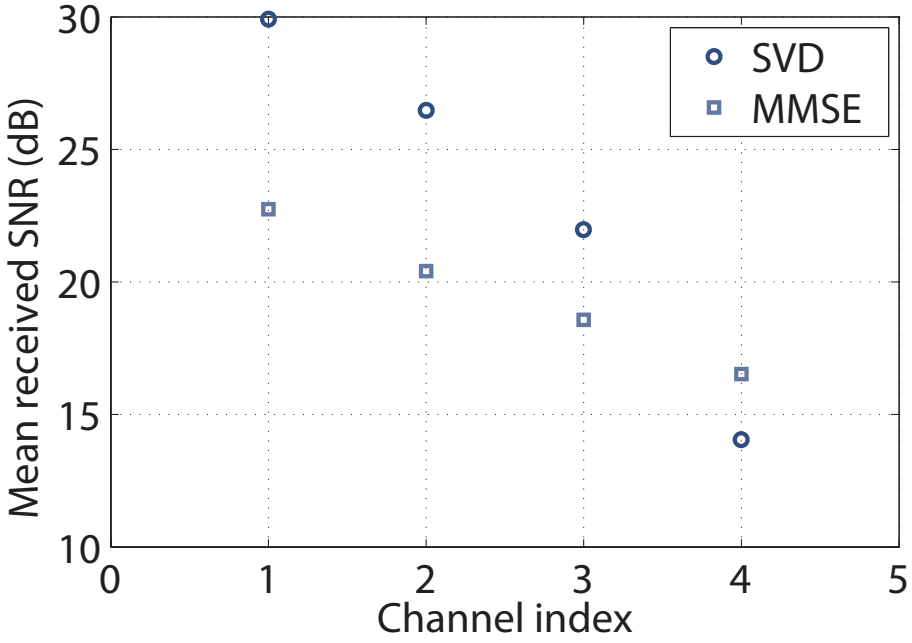


Figure 3.17: Mean received SNRs in a 4×4 system at 20dB.

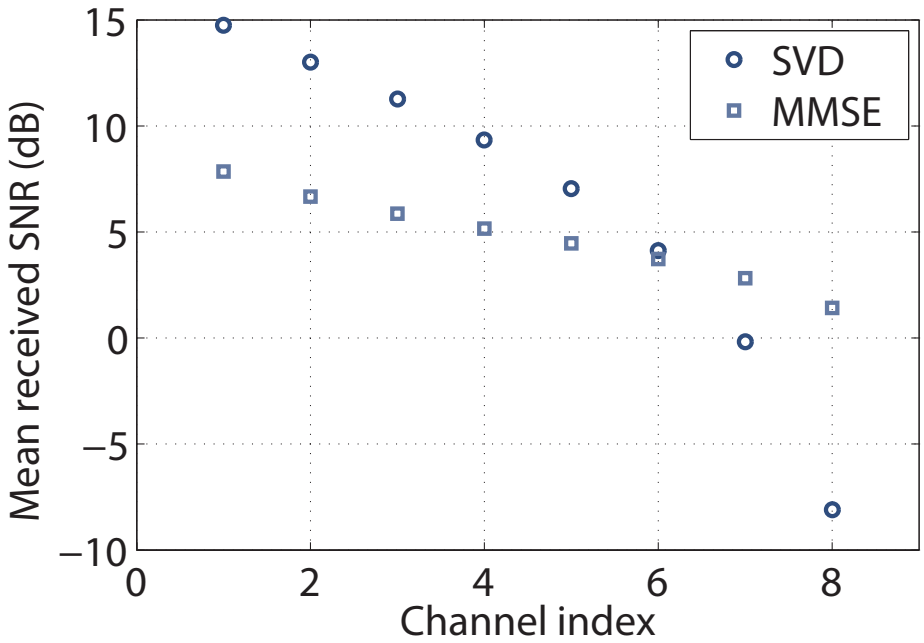


Figure 3.18: Mean received SNRs in a 8×8 system at 1dB.

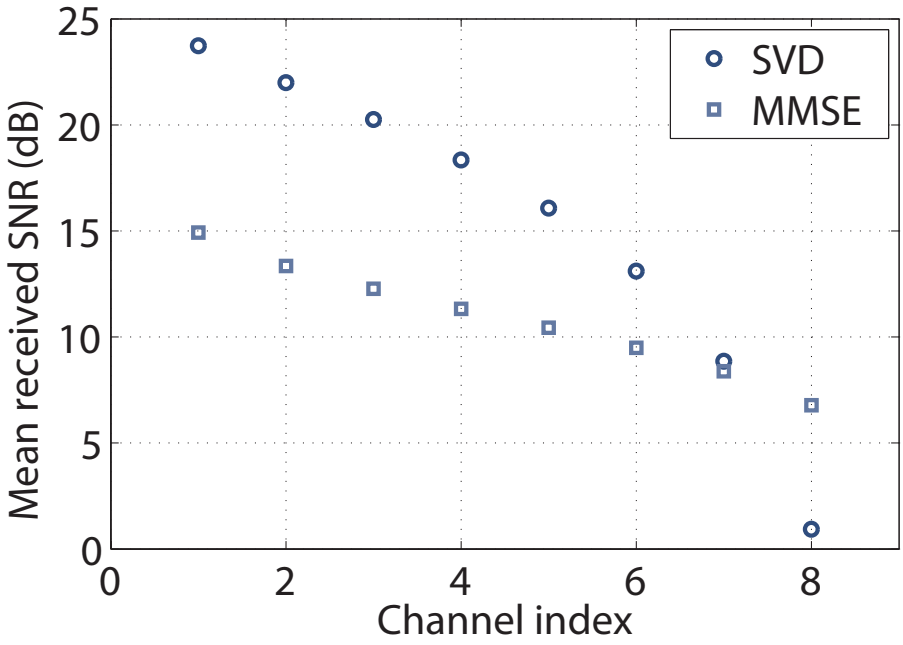


Figure 3.19: Mean received SNRs in a 8×8 system at 10dB.

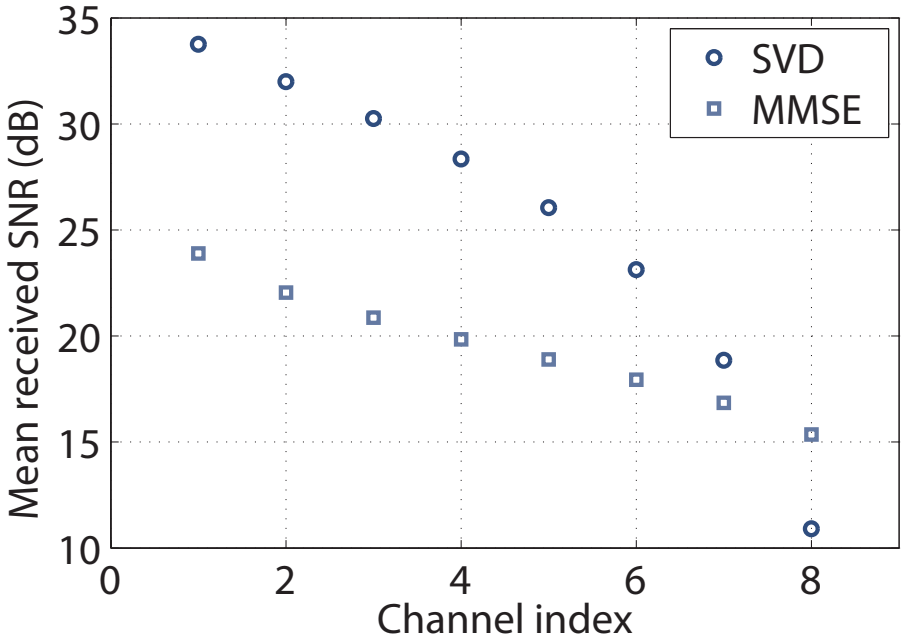


Figure 3.20: Mean received SNRs in a 8×8 system at 20dB.

This implies that as the system size grows the performance difference between the SVD and MMSE system gets bigger. This is due to the ability of the very highest SNRs to reach high level modulations. The effects can be observed in Fig. 3.8. The throughput of the MMSE system can reach about 80% of the SVD system at high SNRs for all the symmetric MIMO systems.

The number of bits transmitted over a MIMO matrix channel depends on various factors such as the system size, transceiver architecture, channel model, etc. Another important factor that determines the throughput in an adaptive MIMO system is the target BER specified by the system since this determines the threshold levels of each modulation mode.

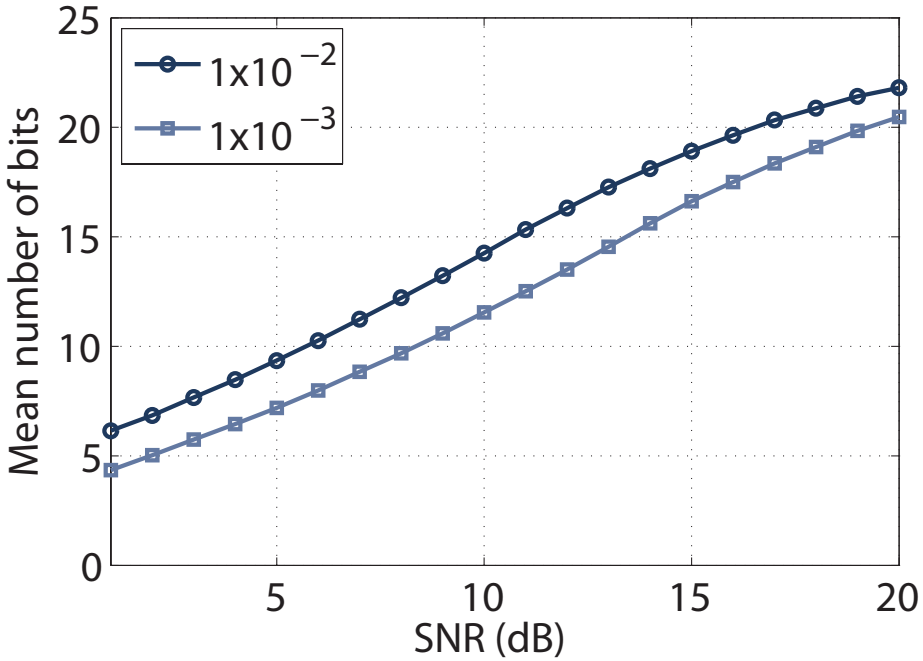


Figure 3.21: Throughputs of a 4×4 SVD system at target BER values of 1×10^{-2} and 1×10^{-3}.

As shown in (2.24), the threshold levels need to be raised in order to provide a better QoS or lower BER. Figure 3.21 shows how the target BER affects the throughput of the system. We can observe that the system with a higher target BER, 1×10^{-2}, transmits slightly more bits than the system with a target BER

of 1×10^{-3}.

In the Rician channel, the reduced rank behavior degrades the quality of the links. Thus, the achievable capacity for spatial multiplexing systems in the Rician channel is relatively low compared to systems in i.i.d Rayleigh environments under the same SNR [17]. Recall that as the Rician factor, K, approaches zero, the Rician channel eventually becomes the Rayleigh channel. This implies that higher values of K lead to more reduced rank behavior. When there is a strong LOS component, e.g., K = 8dB, which is typically assumed in suburban areas, the throughput is considerably reduced, especially in the higher SNR region as shown in Fig. 3.22.

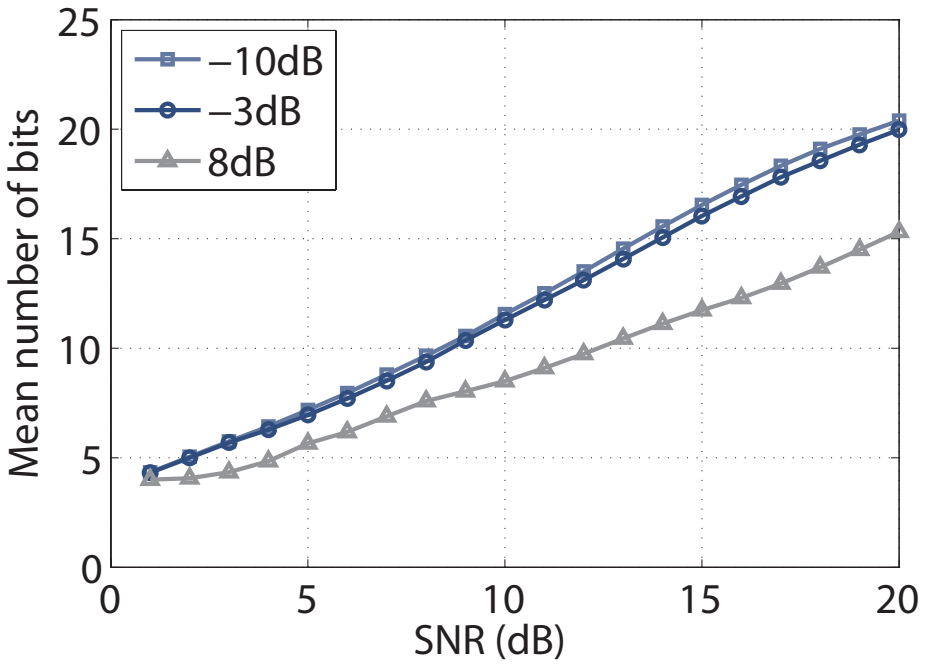


Figure 3.22: Number of bits transmitted over Rician channels with Rician factor K = -10dB, -3dB, and 8dB in a 4×4 SVD system. Target BER = 1×10^{-3}.

Spatially correlated channels are also said to be rank-deficient as there may be only a few dominant eigenmodes while the other eigenvalues are relatively weak. The standard model of a Rayleigh fading channel with spatial correlation

at both TX and RX is given in (2.7) where the transmit correlation matrix, $\mathbf{R}_{\text{tx}}$, and the receive correlation matrix, $\mathbf{R}_{\text{rx}}$, are given in (2.8) and (2.9). The transmit correlation coefficient, α, and the receive correlation coefficient, β, vary between 0 and 1 where 0 indicates that the channels are uncorrelated and 1 means that the channels are completely correlated. As depicted in Fig. 3.23, more correlation between adjacent antennas degrades the performance of the system.

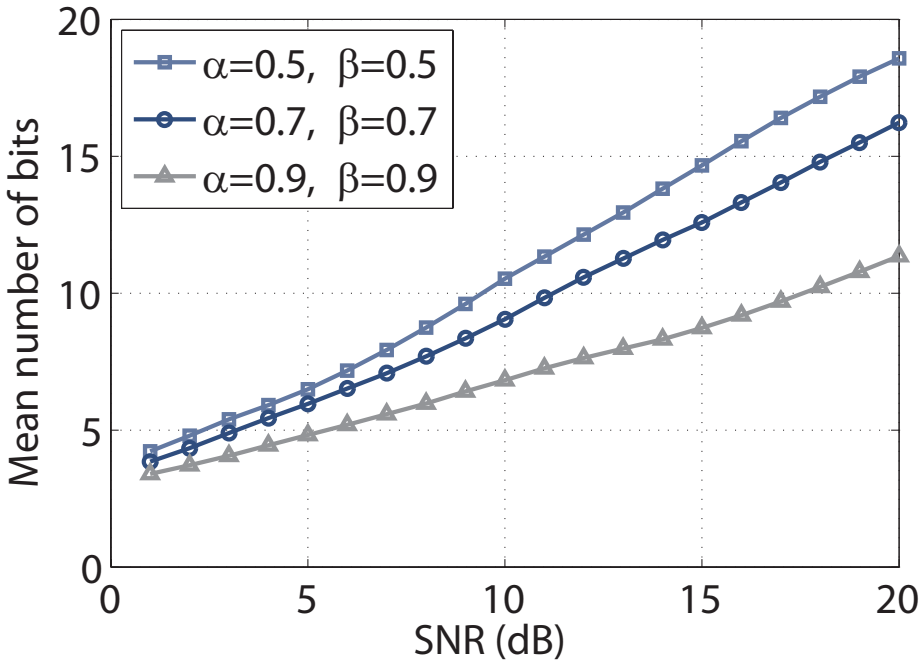


Figure 3.23: Number of bits transmitted over correlated channels with TX correlation coefficient α and RX correlation coefficient β in a 4×4 SVD system. Target BER $= 1 \times 10^{-3}$.

3.6 Summary

Adaptive modulation and power allocation schemes enhance the performance of MIMO systems. Chapter 3 presented a brief power allocation theory and three power allocation algorithms.

The fullsearch algorithm, that ensures the resultant power scaling factors are optimal, considers all N^m possible power scaling combinations where N is the number of modulation modes and m is the number of channels. Note that the number of power scaling combinations that the fullsearch algorithm has to check increases exponentially as the number of channel, m, increases. As shown in Tables 3.17 - 3.19, its high computational workload makes it almost inapplicable in practical applications.

In contrast, the fast and the modified fast algorithms presented in this chapter reduces the mean processing time significantly while achieving, on average, around 97.6% and 99.3% agreement with the optimal results respectively. Note that the modified fast algorithm shows a better performance than the fast method with a slightly longer processing time.

Chapter 4

Excess Power Allocation

Chapter 3 showed three algorithms for power allocation under power and BER constraints. After the power allocation procedure, there is always some power that remains unused. The unused power is called the excess power. The total excess power, E, is distributed to each active channel to improve the system's instantaneous and temporal behavior by giving an extra buffer to the received SNRs. Thus, SNR_i is scaled to $(e_i + p_i)SNR_i$ where e_i is the excess power assigned to SNR_i and p_i is the power scaling factor of SNR_i. Note that $E = \sum e_i$.

This chapter presents five different methods to allocate the excess power. Comparisons between the excess power allocation methods are given at the end of this chapter.

4.1 Benefits of the Excess Power Allocation

Power allocation scales each received SNR to the corresponding threshold level as discussed in Chapter 3. Then, the excess power is used to further scale up the SNRs above the threshold level as depicted in Fig. 4.1. By scaling the SNRs above the threshold level, the instantaneous performance of the system improves since the threshold levels are the minimum required SNR to satisfy the target BER using the corresponding modulation mode. Thus, each channel

achieves a lower BER than the target BER with the excess power.

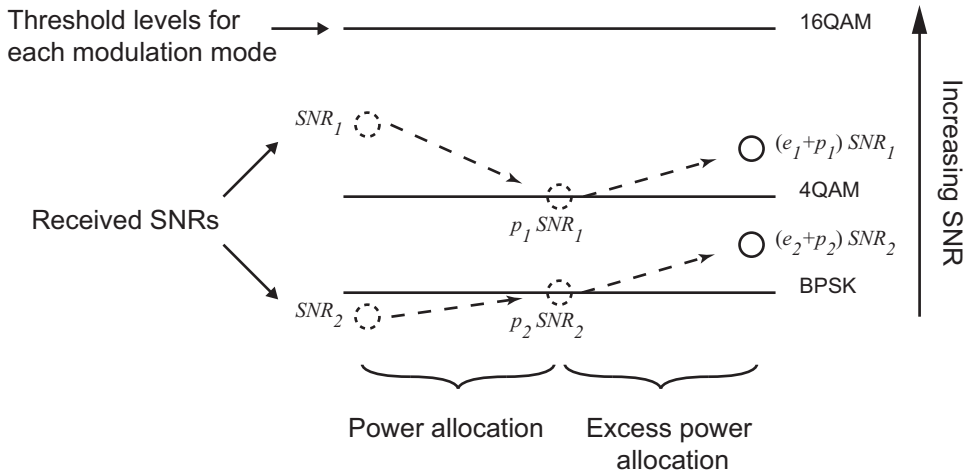


Figure 4.1: Illustration of excess power allocation in a 2×2 system.

Another benefit of the excess power allocation is the improved temporal behavior of the system. Figure 4.2 shows a comparison of the temporal behavior of a 2×2 system with (solid line) and without (dashed line) the excess power.

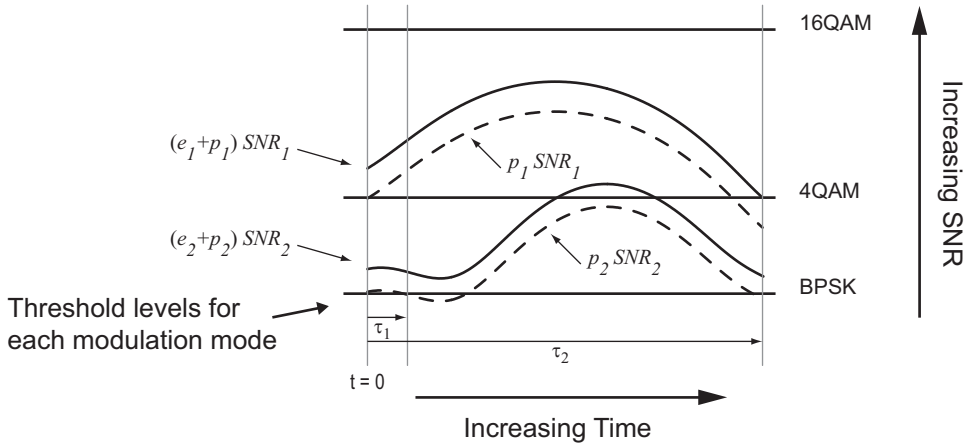


Figure 4.2: Improved temporal behavior of a 2×2 system.

As depicted in Fig. 4.2, without the excess power SNR_2 goes below the BPSK threshold level when $t = \tau_1$. Once the SNR has fallen below the threshold, the BER is greater than the target BER if the same modulation is used.

Hence, in order to satisfy the target BER, new power allocation must be performed.

However, when the excess power is used, the system can hold the same power scaling factors without sacrificing QoS for τ_2 seconds which is significantly longer than τ_1. Note that SNR_1 goes below the threshold level of 4QAM when $t = \tau_2$. Hence, new power allocation is required at $t = \tau_2$. Therefore, the amount of computational workload is reduced since the number of performing power allocation is reduced.

Figure 4.2 is slightly misleading since an SVD system must constantly update its CSI to perform TX and RX processing. However, it is still possible that channel estimation, feedback and adaptive modulation could occur less regularly if the excess power is judiciously used to improve BERs. Hence, we consider the length of time that the BER thresholds are maintained as a guide to the performance of the excess power allocation methods.

4.2 Equal Increment Above the Thresholds

This method of excess power allocation distributes the total excess power, E, so that the resultant SNRs are equally increased above the thresholds. In other words, the difference, d_i, between the scaled SNR, $(e_i + p_i)SNR_i$, and the corresponding threshold level, $p_i SNR_i$, is equal for all the SNRs as shown in Fig. 4.3.

Power allocation scales the SNRs to $p_i SNR_i$, which is a threshold level of a modulation mode. Hence, the difference is defined as

$$\begin{aligned} d_i &= (e_i + p_i)SNR_i - p_i SNR_i, \\ &= e_i SNR_i. \end{aligned} \tag{4.1}$$

In the equal increment method, the diferences are equal for all active SNRs. Hence, we can formulate the following relationship.

$$e_1 SNR_1 = e_2 SNR_2 = \cdots = e_n SNR_n, \tag{4.2}$$

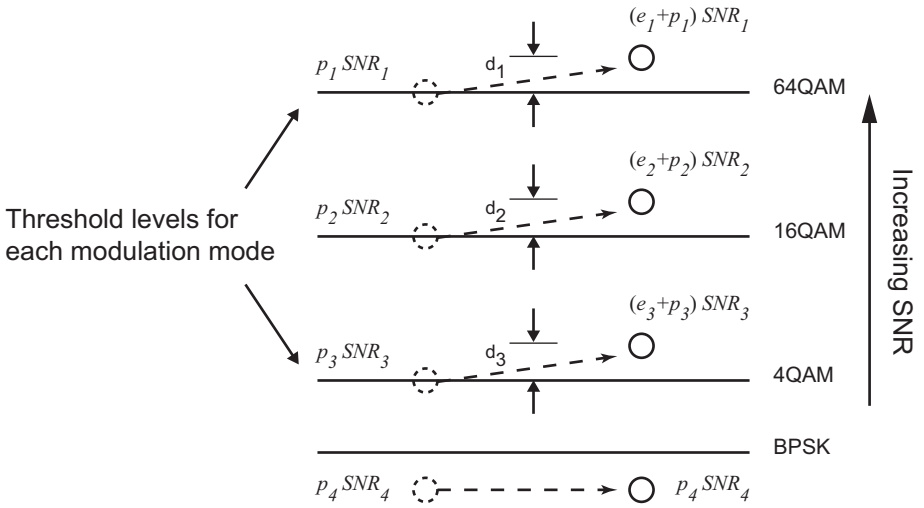


Figure 4.3: Illustration of the equal increment method of excess power allocation. The differences, d_i's, are all equal.

where n is the number of active channels. By normalizing e_1 to 1, (4.2) can be re-written as

$$\overline{e}_1 SNR_1 = \overline{e}_2 SNR_2 = \cdots = \overline{e}_n SNR_n, \tag{4.3}$$

where $\overline{e}_i$ represent the normalized excess power. Thus, the normalized excess powers are given by

$$\begin{aligned} \overline{e}_2 &= SNR_1/SNR_2, \\ &\vdots \\ \overline{e}_n &= SNR_1/SNR_n. \end{aligned} \tag{4.4}$$

Hence, we can compute the excess powers of each SNR by solving

$$e_i = \frac{E}{\sum \overline{e}_i} \overline{e}_i. \tag{4.5}$$

The final power scaling factors, (p_i+e_i), scale up the SNRs an equal amount above the threshold levels.

4.3 Equal BER Improvement

The excess power is used to improve the BER performance of the system so that the BER of each active channel improves equally. We approach this problem by

Modulation Mode	α	β
4QAM	1/2	1/2
16QAM	3/8	1/10
64QAM	7/24	1/42

Table 4.1: α and β values for the QAM modulation modes.

considering it as equivalent to raising the thresholds or alternatively, reducing the target BER.

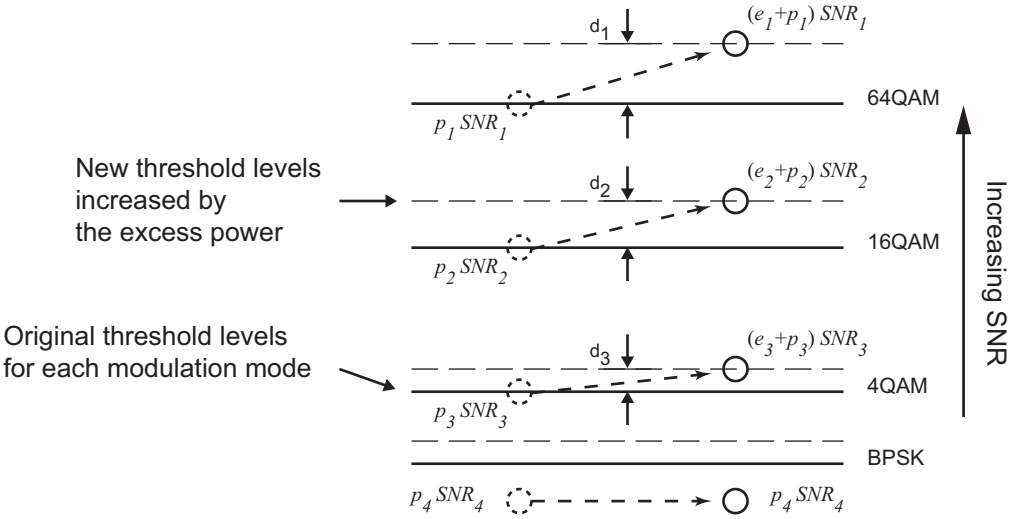


Figure 4.4: Illustration of the equal BER improvement method of excess power allocation.

With the BER formulae given in (2.19) and (2.21), we can calculate how the excess power affects the BER. To find the new threshold levels, we first need to find the new BER. As we aim to scale the SNRs to $(p_i + e_i)SNR_i$, we can re-write (2.19) and (2.21) as

$$(p_i + e_i)\, SNR_i \approx \left(\text{erfc}^{-1}\left(2\text{BER}_{\text{new}}\right)\right)^2, \tag{4.6}$$

and

$$(p_i + e_i)\, SNR_i \approx \frac{1}{\beta_i}\left(\text{erfc}^{-1}\left(\frac{\text{BER}_{\text{new}}}{\alpha_i}\right)\right)^2, \tag{4.7}$$

where α and β are defined in (2.22) and (2.23) respectively. The values of α and β for the QAM modulations are given in Table 4.1.

Note that (4.6) only holds for the SNRs using BPSK and (4.7) only holds for SNRs using square MQAM. Thus, in the equal BER improvement method, we have to consider the two cases separately.

For the SNRs using BPSK, the overall power scaling factor, $p_i + e_i$, is given by

$$(p_i + e_i) \approx \frac{1}{SNR_i} \left(\text{erfc}^{-1}\left(2\text{BER}_{\text{new}}\right)\right)^2. \tag{4.8}$$

For the SNRs using higher order MQAM, the overall power scaling factor, $p_i + e_i$, is given by

$$(p_i + e_i) \approx \frac{1}{\beta_i SNR_i} \left(\text{erfc}^{-1}\left(\frac{\text{BER}_{\text{new}}}{\alpha_i}\right)\right)^2. \tag{4.9}$$

Assuming the average transmit symbol energy is one, the sum of all power scaling factors must equal the number of channels, m. Hence, solving

$$\sum (p_i + e_i) - m = 0, \tag{4.10}$$

the new BER can be computed, where $(p_i + e_i)$ is defined in (4.8) and (4.9) for SNRs using BPSK and higher order MQAM modulation modes respectively. The formulae for $(p_i + e_i)$ given in (4.8) and (4.9) involve the erfc function and cannot be easily inverted. Thus, the new BER must be computed numerically. Note that (4.10) is a function of m, α_i, β_i, SNR_i and BER_{new}. Since m, α_i, β_i and SNR_i are known, it is a function of the single unknown, BER_{new}, which leads to a simple one-dimensional numerical solution.

A simple root-finding algorithm to find BER_{new} is the bisection method which is robust but relatively slow. The bisection method requires two initial points, a and b, such that $f(a)$ and $f(b)$ have opposite signs. In the excess power allocation, we know that the new BER must lie between the target BER and one hundredth of the target BER. For example, if the target BER is 1×10^{-3}, we can use 1×10^{-3} and 1×10^{-5} as the initial values. Thus, the interval $[1 \times 10^{-3}, 1 \times 10^{-5}]$ is halved until it reaches the root in an iterative manner. More details of the bisection method are available in many mathematics text

System Size (N_r, N_t)	λ_1	λ_2	λ_3	λ_4
(2,2)	3.25	0.25		
(2,4)	5.4023	1.0273		
(3,3)	5.5135	1.1385	0.1111	
(4,4)	7.6392	2.2442	0.5964	0.0625

Table 4.2: Eigenvalue variances [1]

books [31]. Most packages also have built-in optimization and root-finding techniques which will handle this problem.

After the new BER is found, the corresponding new threshold levels are computed as in (2.24). Then, the power scaling factors, $(p_i + e_i)$, can be computed easily.

4.4 Excess Power Distributed Proportional to the Eigenvalue Variances

This method of excess power allocation distributes the total excess power, E, proportional to the eigenvalue variances. The eigenvalue variances show how variable the received SNRs are. Thus, this method provides a larger buffer for more variable channels. The excess powers are allocated to the channels proportional to the eigenvalue variances given in [1]. These eigenvalue variances are also listed in Table 4.2 where $\lambda_1 > \cdots > \lambda_4$.

Note that the higher SNR varies more rapidly. Hence, more excess power is given to the higher SNR. Since the total excess power is given to each SNR proportional to the variance, the excess power given to each SNR can be formulated as

$$e_i = \frac{E}{\sum_{i=1}^{n} \mathrm{Var}\left(\lambda_i\right)} \mathrm{Var}\left(\lambda_i\right), \tag{4.11}$$

where n is the number of active channels.

4.5 Excess Power Distributed Proportional to the Received SNRs

Excess power allocation proportional to the received SNRs is a technique inspired by the fact that the higher SNRs are changing more rapidly compared to the lower SNRs. Thus, this method distributes the total excess power, E, in the following way:

$$e_i = \frac{E}{\sum_{i=1}^{n} SNR_i} SNR_i. \tag{4.12}$$

This method shows similar results to the method using eigenvalue variances.

4.6 The Gradient Method

The gradient method distributes the total excess power, E, to each SNR proportional to the difference between the current SNR and the SNR that was received τ seconds before. The difference is computed as

$$\Delta SNR_i = d_i = SNR_i(t-\tau) - SNR_i(t), \tag{4.13}$$

where $SNR_i(t)$ denotes the received SNR_i at time t.

If the SNR is increasing, i.e., $\Delta SNR_i < 0$, no excess power is given to the SNR. The excess power is only given to the SNRs which are decreasing and for these, the allocation is proportional to the amount of the decrement. Thus, the excess power for each SNR is computed as

$$e_i = \begin{cases} 0, & \text{for } d_i \leq 0 \\ \frac{E}{D} d_i, & \text{for } d_i > 0 \end{cases}, \tag{4.14}$$

where E is the total excess power and D is the sum of the d_i's that are greater than 0.

This method assumes that the increasing channels keep increasing and decreasing channels keep decreasing. Thus, more excess power is given to

an SNR that is decreasing rapidly relative to other SNR that are decreasing gradually and no excess power is allocated to the increasing SNRs.

In the case that all the SNRs are increasing, the total excess power is distributed evenly to the active channels.

4.7 A Comparison of Excess Power Allocation Methods

To compare the proposed excess power allocation methods, a 4×4 SVD system is considered in a Rayleigh fading environment. To model the temporal behavior of the Rayleigh fading channel, the Jakes' model is used and the ACF of the channel coefficients is given by (2.5). The scope of this simulation was to determine the most robust excess power allocation method in the time varying channel. The time that each method can hold the current modulation mode without loss of QoS, is measured in milli-seconds. Some statistical simulation results are shown in Tables 4.3- 4.8. Note that the following abbreviations are used to denote the excess power allocation methods.

EI	Equal increment method
EB	Equal BER improvement method
EV	Eigenvalue variance method
RS	Received SNR method
GR	Gradient method

Each set of simulation results is obtained by running 1,000 time varying channels at various Doppler frequency, f_D, values : 10Hz, 50Hz and 100Hz. Since the simulation is performed on a 4×4 system, the maximum number of active channels is 4. However, according to the channel condition, the number of active channels may vary. The ratio given in the tables describes how often

No. of active Ch.	EI	EB	EV	RS	GR	spare	Ratio
2	109.5	114.7	114.2	114.8	117.7	0.9391	2.2%
3	18.2	20.8	19.1	17.5	16.5	0.7162	76.4%
4	3.6	5.5	3.9	3.7	4.0	0.5173	21.4%

Table 4.3: Mean hold time (ms) of the excess power allocation methods for various numbers of active channels. Maximum Doppler shift, f_D, is 10Hz.

	EI	EB	EV	RS	GR
Maximum	1712.3	1712.3	1711.0	1710.8	1711.5
Upper Quartile	13.3	17.4	16.5	16.1	14.6
Median	4.6	6.7	5.5	4.5	5.4
Lower Quartile	1.6	2.2	1.5	1.0	1.5
Minimum	1	1	1	1	1
Standard Deviation	62.6	63.7	62.8	60.9	59.2
Mean	17.1	19.6	18.0	16.7	16.1

Table 4.4: Statistical information on the hold time (ms) of the excess power allocation methods. Maximum Doppler shift, f_D, is 10Hz.

that number of channels is active and the spare is the mean value of the total excess power.

From the simulation results, there is no noticeable difference between the excess power allocation methods. Even though, the equal BER improvement method seems logically the best method for excess power allocation, this method has a fairly high complexity and computational workload since this method involves a bisection root-finding algorithm. Thus, further research on the equal BER improvement method is suggested. One suggested method to simplify the equal BER method is to use an approximate function to compute

No. of active Ch.	EI	EB	EV	RS	GR	spare	Ratio
2	25.0	23.3	39.0	26.7	15.6	0.8796	1.3%
3	3.0	4.1	3.8	3.5	3.2	0.7148	76.2%
4	0.8	1.2	0.9	0.8	0.8	0.0.5277	22.5%

Table 4.5: Mean hold time (ms) of the excess power allocation methods for various numbers of active channels. Maximum Doppler shift, f_D, is 50Hz.

	EI	EB	EV	RS	GR
Maximum	104.3	136.9	220.3	121.7	91.3
Upper Quartile	2.7	3.5	3.25	3.0	2.8
Median	0.9	1.4	1.1	0.9	0.9
Lower Quartile	0.3	0.6	0.3	0.2	0.2
Minimum	1	1	1	1	1
Standard Deviation	6.4	8.8	10.6	8.0	6.1
Mean	2.8	3.7	3.6	3.2	2.8

Table 4.6: Statistical information on the hold time (ms) of the excess power allocation methods. Maximum Doppler shift, f_D, is 50Hz.

No. of active Ch.	EI	EB	EV	RS	GR	spare	Ratio
2	8.3	10.0	8.8	8.9	4.3	0.8780	1.3%
3	2.0	2.3	2.2	2.0	1.6	0.7148	77.0%
4	0.4	0.5	0.4	0.4	0.3	0.5320	21.3%

Table 4.7: Mean hold time (ms) of the excess power allocation methods for various numbers of active channels. Maximum Doppler shift, f_D, is 100Hz.

	EI	EB	EV	RS	GR
Maximum	83.1	68.8	68.6	36.8	29.7
Upper Quartile	1.5	2.05	1.9	1.9	1.2
Median	0.5	0.7	0.7	0.6	0.35
Lower Quartile	0.2	0.3	0.2	0.15	0.1
Minimum	1	1	1	1	1
Standard Deviation	4.2	4.1	4.0	3.1	2.8
Mean	1.7	2.0	1.9	1.7	1.3

Table 4.8: Statistical information on the hold time (ms) of the excess power allocation methods. Maximum Doppler shift, f_D, is 100Hz.

the inverse of the complementary error function, such as [32]

$$\text{erfc}^{-1}(x) \leq \sqrt{-\ln(x)}. \tag{4.15}$$

By using the approximate function, the bisection root-finding algorithm can be avoided. Hence, (4.6) and (4.7) can be re-written as

$$(p_i + e_i)SNR_i = \begin{cases} -\ln\left(2\text{BER}_{\text{new}}\right), & \text{for BPSK} \\ -\frac{1}{\beta_i}\ln\left(\frac{\text{BER}_{\text{new}}}{\alpha_i}\right), & \text{for 4QAM, 16QAM, 64QAM} \end{cases} \tag{4.16}$$

Then, (4.10) can be solved without bisection algorithm. However, using this simple approximation may not be accurate enough. For example, if the received SNRs in a 2×2 system are 26.0489 and 4.8149, the power allocation process will scale the first SNR to 4QAM mode and the latter to BPSK mode. If the target BER is 1×10^{-3}, the corresponding power scaling factors are 0.3666 and 0.9917 respectively which gives 0.6417 as the excess power. By the bisection root-finding algorithm, the new improved BER is 8.8468×10^{-5} whereas the approximation method result in a larger BER as shown below. With the approximation in (4.15), we can re-write (4.16) as

$$(p_i + e_i)SNR_i = \begin{cases} -\ln\left(2\text{BER}_{\text{new}}\right), & \text{for BPSK} \\ -2\ln\left(2\text{BER}_{\text{new}}\right), & \text{for 4QAM} \end{cases} \tag{4.17}$$

since $\alpha = 1/2$ and $\beta = 1/2$ for 4QAM as given in Table 4.1. To solve for BER_{new}, we sum both sides as

$$\begin{aligned} \sum(p_i + e_i) &= \frac{-\ln(2\text{BER}_{\text{new}})}{4.8149} + \frac{-2\ln(2\text{BER}_{\text{new}})}{26.0489} \\ &= 2 \end{aligned} \tag{4.18}$$

$$\tag{4.19}$$

Hence,

$$-\ln\left\{\left(2\text{BER}_{\text{new}}\right)^{1/4.8149}\right\} - \ln\left\{\left(2\text{BER}_{\text{new}}\right)^{2/26.0489}\right\} = 2. \tag{4.20}$$

Combining the logarithms gives

$$-\ln\left\{2^{1/4.8149}2^{2/26.0849}\left(\mathrm{BER}_{\mathrm{new}}\right)^{0.28447}\right\}=2, \tag{4.21}$$

and solving (4.21) gives

$$\mathrm{BER}_{\mathrm{new}}=4.422\times10^{-4} \tag{4.22}$$

The agreement between the exact solution, 8.85×10^{-5}, and the closed form approximation, 4.42×10^{-4}, is not precise but may still be useful. For example, if the approximate $\mathrm{BER}_{\mathrm{new}}$ is too high then there is still some unused power. This can be allocated evenly. Similarly, if the $\mathrm{BER}_{\mathrm{new}}$ is too low, too much power is required. In this situation, power can be removed evenly from each channel. This simple approach may give approximately the same results as the exact technique without the need for a numerical solution. The general approximate solution is outlined below.

If BPSK is used for $k\leq m$ channels then

$$\sum_{i=1}^{k}\frac{1}{SNR_i}\left(\mathrm{erfc}^{-1}\left(2p\right)\right)^2+\sum_{i=k+1}^{m}\frac{1}{\beta_i SNR_i}\left(\mathrm{erfc}^{-1}\left(\frac{p}{\alpha_i}\right)\right)^2=m, \tag{4.23}$$

where p is the probability of error. Using the $\mathrm{erfc}^{-1}(\cdot)$ approximation we have

$$\sum_{i=1}^{k}\frac{-\ln(2p)}{SNR_i}+\sum_{i=k+1}^{m}\frac{-\ln(p/\alpha_i)}{\beta_i SNR_i}=m. \tag{4.24}$$

Defining $\alpha_1=\alpha_2=\cdots=\alpha_k=1/2$ and $\beta_1=\beta_2=\cdots=\beta_k=1$, we have

$$\begin{aligned}\sum_{i=1}^{m}\frac{-\ln(p/\alpha_i)}{\beta_i SNR_i}&=m\\ \sum_{i=1}^{m}\frac{\ln(\alpha_i)}{\beta_i SNR_i}-m&=\sum_{i=1}^{m}\frac{\ln(p)}{\beta_i SNR_i}.\end{aligned} \tag{4.25}$$

Solving (4.25) gives the answer

$$p=\exp\left\{\sum_{i=1}^{m}\frac{\ln(\alpha_i)}{\beta_i SNR_i}-m\right\}^{1/\gamma}, \tag{4.26}$$

where

$$\gamma = \sum_{i=1}^{m} \frac{1}{\beta_i SNR_i}. \tag{4.27}$$

Note that the same solution holds when no BPSK modulation is used ($k = 0$) since here the special definition of $\alpha_1, \cdots, \alpha_k$ and $\beta_1, \cdots, \beta_k$ are not required and all the parameters relate to the relevant QAM modulation.

4.8 Summary

The power allocation process scales the received SNRs up or down so that the SNRs are just strong enough to achieve the target BER for the corresponding modulation modes. Hence, after the power allocation procedure, there is always some power that remains unused, called the excess power, E. Excess power allocation is used to use the power, a precious resource, more wisely.

Excess power allocation distributes the excess power, E, to each active channel to improve the system's instantaneous and temporal behavior. Benefits of the excess power allocation are presented in Section 4.1, and Sections 4.2 - 4.6 proposed five different methods of excess power allocation. Some statistical simulation results given in Section 4.7 showed that all five different methods provide similar improvement to the system's instantaneous and temporal behavior. Even though, the equal BER method showed a slightly better performance than other methods, this method involves a bisection root-finding algorithm which increases computational workload and complexity. One approach to simplify the equal BER method is suggested at the end of Section 4.7.

Chapter 5

Conclusion and Future Work

In this chapter, we draw conclusions based on the work presented in this thesis and summarize the major achievements and contributions. Some suggestions for future work are also presented at the end of the chapter.

5.1 Conclusions

A wireless communication architecture with multiple antenna elements at both transmitter and receiver, known as a MIMO system, is a promising approach to achieve high capacity for next generation wireless technologies. Recent research has shown that adaptive modulation and power allocation techniques can be used to further improve the performance of the system. Various factors affect the capacity of the system, such as the system size, transceiver architecture, channel model, and the target BER. In Chapter 2, the potential benefits of MIMO and the commonly used channel models: the Rayleigh, the Rician, and spatially correlated Rayleigh channel model, are elaborated as well as a brief description of adaptive modulation and power allocation.

In Chapter 3, we concentrated on developing new power allocation algorithms for MIMO systems. For a target BER specified for a MIMO system, the required SNR or the threshold level of each modulation is computed by (2.19) for the BPSK mode and (2.21) for the 4QAM, 16QAM, and 64QAM

modes. The power scaling factors found by the power allocation algorithm are used to scale the received SNRs to the threshold level of the corresponding modulation mode to maximize the throughput. By scaling the received SNRs to certain modulation modes, the throughput of the system increases under power and QoS constraints.

The fullsearch algorithm proposed in this thesis computes the optimal power scaling factors by considering all N^m possible combinations where N is the number of modulation modes and m is the number of channels. This algorithm ensures that the found power scaling factors are optimal. However, this algorithm also considers invalid combinations which increases the computational workload especially in the low SNR region. As shown in Table 3.1, the invalid ratio is high in the low SNR region and gradually decreases as the SNR increases. As a result of the decreased invalid ratio values at high SNRs, the fullsearch algorithm must compare more valid combinations which consequently increases its computing time as shown in Tables 3.17 - 3.19. Note that the fullsearch algorithm mean run-time increases nearly 10-fold from 2×2 to 4×4 whereas the fast and the modified fast algorithm only increase by about 20%. In a 6×6 system, the mean cpu time taken by the fullsearch algorithm is about 2300 times longer than the fast algorithm at 20dB. Since, the fullsearch algorithm has a high computational workload, it is almost inapplicable for practical applications.

In Chapter 3, we also presented the fast algorithm and the modified fast algorithm. The fast and the modified fast algorithm reduce the power allocation processing time significantly while achieving, on average, around 97.6% and 99.3% agreement with the fullsearch (optimal) results respectively. Moreover, unlike the fullsearch algorithm, the fast and the modified fast algorithm do not show any relationship between the SNR and the processing time. Since the fast algorithm can achieve around 97.6% of the optimal throughput on average within a significantly less time compare to the fullsearch algorithm, one would

pick the fast algorithm as the best power allocation algorithm. Even though the modified fast algorithm can achieve higher performance, the improvement is barely noticeable whereas the modified fast algorithm takes about 0.1 ms longer on average to compute the power scaling factors. The following graphical results are presented at the end of Chapter 3:

- Throughput comparisons between the power allocation algorithms for 2×2 and 4×4 SVD systems, and 2×2 and 4×4 MMSE systems in the Rayleigh fading channel with target BER $= 1 \times 10^{-3}$.
- Performance comparisons between SVD systems and MMSE systems for 2×2, 4×2, and 4×4 systems with target BER $= 1 \times 10^{-3}$.
  In an asymmetric configuration, performance of the MMSE system is almost as good as the SVD system.
- Mean received SNR comparisons between SVD systems and MMSE systems for 2×2, 4×2, 4×4, and 8×8 systems.
- Effect of target BER on the throughput of the system.
  Higher target BER induces higher throughput due to the decreased threshold level for modulation modes.
- Effect of the Rician factor, K, on the throughput of the system.
  Strong LOS components degrade the throughput of the system.
- Effect of correlation coefficients, α and β, on the throughput of the system.
  As the correlation coefficients approach 1, i.e., as the correlated between adjacent antennas increases, the throughput of the system is reduced.

After the power allocation procedure is complete, any excess power is used to improve the instantaneous and temporal behavior of the system as discussed in Chapter 4. Some statistical simulation results are provided at the end of Chapter 4. The simulation results show that the temporal performance of the proposed excess power allocation methods are similar. However, the equal BER

improvement method showed a slightly better performance. When the maximum Doppler shift, f_D, is 10Hz, the equal BER improvement method could hold the assigned power scaling factors for about 19.6 ms without sacrificing the QoS. As the maximum Doppler shift increases, the channel coefficients vary more rapidly. Hence, the hold time decreases simultaneously. Thus, at $f_D = 100$Hz, the mean hold time decreases to 2.0 ms.

5.2 Suggestions for Future Work

Power allocation is still an active area of research. This thesis introduced the fullsearch algorithm, the fast algorithm and the modified fast algorithm which can be used in a MIMO system to further improve the performance of the system. In particular, the fast algorithm achieves a fairly high performance in a relatively short period. Since the fast algorithm is very close to optimal and far more efficient, a detailed investigation of the fast algorithm is warranted. This work could attempt further fine tuning of the algorithm to improve computation speed. Also, the use of efficient algorithms to perform the computations could be considered as part of the pathway to actual implementation.

In Chapter 4, we showed that the equal BER improvement method is the best method to distribute the excess power to each SNR to improve the instantaneous and temporal behavior of the system. However, its computational workload is an issue in practical use. Thus, a simplified version of the equal BER improvement method is required. In this thesis, we have proposed a method which partially solves this problem. Further work is necessary to fully evaluate this technique. Also, more accurate approximations to the inverse error function could be considered. Mathematically, this is challenging since the approximation needs to be accurate enough to give improved results while simple enough to lead to a closed form solution to the power allocation.

Bibliography

[1] P. H. Kuo, "Channel variations in MIMO wireless communication systems: Eigen-structure perspectives," Ph.D. dissertation, University of Canterbury, Jun. 2007.

[2] S. Acharya and S. Teltscher, "Press release: New ITU ICT development index compares 154 countries northern Europe tops ICT developments," [Online], Mar. 2009, Available: http://www.itu.int/newsroom/press-releases/2009/07.html [Accessed: Sept. 01, 2009].

[3] S. Haykin, *Communication Systems*, 4th ed. New York: John Wiley & Sons, Inc, 2001.

[4] G. J. Foschini and M. J. Gans, "On limits of wireless communications in a fading environment when using multiple antennas," *Wireless Pers. Commun.*, vol. 6, no. 3, pp. 311–335, Mar. 1998.

[5] I. E. Telatar, "Capacity of multi-antenna Gaussian channels," *Eur. Trans. Telecomm.*, vol. 10, no. 6, pp. 585–595, Nov. 1999.

[6] Y. H. Pan, K. B. Letaief, and Z. Cao, "Dynamic resource allocation with adaptive beamforming for MIMO/OFDM systems under perfect and imperfect CSI," in *Proc. WCNC IEEE*, vol. 1, Atlanta, Georgia, USA, Mar. 2004, pp. 93–97.

[7] A. Feiten, R. Marthar, and M. Reyer, "Rate and power allocation for multiuser OFDM: An effective heuristic verified by branch-and-bound," *IEEE Trans. Commun.*, vol. 7, no. 1, pp. 60–64, Jan. 2008.

[8] F. Rey, M. Lamarca, and G. Vazquez, "Robust power allocation algorithms for MIMO OFDM systems with imperfect CSI," *IEEE Trans. Sig. Processing*, vol. 53, no. 3, pp. 1070–1085, Mar. 2005.

[9] A. Pandharipande, "Adaptive modulation for MIMO-OFDM systems," in *Proc. IEEE VTC*, Los Angeles, CA, USA, Sept. 2004, pp. 1266–1270.

[10] S. T. Chung and A. J. Goldsmith, "Degrees of freedom in adaptive modulation: a unified view," *IEEE Trans. Commun.*, vol. 49, no. 9, pp. 1561–1571, Sept. 2001.

[11] G. G. Raleigh and J. M. Cioffi, "Spatio-temporal coding for wireless communication," *IEEE Trans. Commun.*, vol. 46, no. 3, pp. 357–366, Mar. 1998.

[12] X. Zhang and B. Ottersten, "Power allocation and bit loading for spatial multiplexing in MIMO systems," in *Proc. ICASSP IEEE*, Hong Kong, Apr. 2003, pp. 53–56.

[13] X. Jin, H. Jiang, J. Hu, Y. Yuan, C. Zhao, and J. Shi, "Maximum data rate power allocation for MIMO spatial multiplexing systems with imperfect CSI," in *Proc. IEEE VTC*, Barcelona, Spain, Apr. 2009, pp. 1–5.

[14] A. F. Molisch, *Wireless Communications*. New York: John Wiley & Sons Ltd., Jul. 2006.

[15] W. C. Jakes, *Microwave mobile communications*. New York: John Wiley & Sons, Inc., Feb. 1975.

[16] F. R. Farrokhi, G. J. Foschini, A. Lozano, and R. A. Valenzuela, "Link-optimal space-time processing with multiple transmit and receive antennas," *IEEE Commun. Lett.*, vol. 5, no. 3, pp. 85–87, Mar. 2001.

[17] M. Stoytchev and H. Safar, "Statistics of MIMO radio channel in indoor environments," in *Proc. IEEE VTC*, vol. 3, Rhodes, Greece, Oct. 2001, pp. 1804–1808.

[18] G. Lebrun, M. Faulkner, M. Shafi, and P. J. Smith, "MIMO Ricean channel capacity," in *Proc. IEEE ICC*, Paris, France, May. 2004, pp. 2939–2943.

[19] M. Chiani, M. Z. Win, and A. Zanella, "On the capacity of spatially correlated MIMO Rayleigh channels," *IEEE Trans. Info. Theory*, vol. 49, no. 10, pp. 2363–2371, Oct. 2003.

[20] P. Kyritsi, C. Cox, R. Valenzuela, and P. W. Wolniansky, "Correlation analysis based on MIMO channel measurements in an indoor environment," *IEEE J. Sel. Areas Commun.*, vol. 21, no. 5, pp. 713–720, Jun. 2003.

[21] I. Berenguer and Z. Wang, "Space-time coding and signal processing for MIMO communications," *J. Comput. Sci. Tech*, vol. 18, no. 6, pp. 689–702, 2003.

[22] J. Adeane, W. Q. Malik, I. J. Wassell, and D. J. Edwards, "Simple correlated channel model for ultrawideband multiple-input multiple-output systems," *IET Microw. Antennas Propag.*, vol. 1, no. 6, pp. 1177–1181, Dec. 2007.

[23] P. J. Smith, S. Roy, and M. Shafi, "Capacity of MIMO systems with semicorrelated flat fading," *IEEE Trans. Info. Theory*, vol. 49, no. 10, pp. 2781–2788, Oct. 2003.

[24] G. Lebrun, S. Spiteri, and M. Faulkner, "Channel estimation for an SVD-MIMO systems," in *Proc. IEEE ICC*, vol. 5, Paris, France, Jun. 2004, pp. 3025–3029.

[25] H. Sempath, P. Stoica, and A. Paulraj, "Generalized linear precoder and decoder design for MIMO channels using the weighted MMSE criterion," *IEEE Trans. Commun.*, vol. 49, no. 12, pp. 2198–2206, Dec. 2001.

[26] S. H. Choi, "Severely fading MIMO channels," Master's thesis, University of Canterbury, Mar. 2007.

[27] D. N. Liu and M. P. Fitz, "Low complexity affine MMSE detector for iterative detection-decoding MIMO OFDM system," *IEEE Trans. Commun.*, vol. 56, no. 1, pp. 150–158, Jan. 2008.

[28] P. Li, D. Paul, R. Narasimhan, and J. Cioffi, "On the distribution of SINR for the MMSE MIMO receiver and performance analysis," *IEEE Trans. Info. Theory*, vol. 52, no. 1, pp. 271–286, Jan. 2006.

[29] M. Kiessling and J. Speidel, "Analytical performance of MIMO MMSE receivers in correlated Rayleigh fading environments," in *Proc. IEEE VTC*, vol. 3, Orlando, Florida, USA, Oct. 2003, pp. 1738–1742.

[30] A. J. Goldsmith and P. Varaiya, "Capacity of flat fading channels with channel side information," *IEEE Trans. Info. Theory*, vol. 43, no. 6, pp. 1986–1992, Nov. 1997.

[31] J. F. Epperson, *An introduction to numerical methods and analysis.* New York: John Wiley & Sons, Inc, 2002.

[32] M. Chiani and D. Dardari, "Improved exponential bounds and approximation for the Q-function with application to average error probability computation," in *Proc. GLOBECOM IEEE*, vol. 2, Taipei, Taiwan, Nov. 2002, pp. 1399–1402.